

N° 2
Septembre
1985
18 F

LISTING : TRAITEMENT DE TEXTE

TEST : UNITE DE DISQUETTES DDI-1

SUPPLEMENT : INTRODUCTION A LA PROGRAMMATION

M 1157 - 02 - 18 F

AMSTRAD MAGAZINE

N° 2
SEPTEMBRE 1985

SOMMAIRE

Directeur de la publication, rédacteur en chef : Jean Kaminsky. **Coordination de la rédaction :** Philippe Lamigeon. **Rédaction :** Daniel Martin, Eric Charton, R.P. Spiegel, Pascal Manoury, Michel Merlet, Michel Lepourcelet. **Secrétaire de rédaction :** Mireille Massonnet. **Maquette :** Marc Soria. **Régie publicitaire :** Néo-Média, 55, avenue Jean-Jaurès, 75019 Paris. Tél. 241.81.81. Jean-Yves Primas. **Commission paritaire :** en cours. **Dépôt légal :** 3e trimestre 1985. **Imprimé par** SNIL-RBI. **Edité par :** Laser Magazine, 55, avenue Jean-Jaurès, 75019 Paris.

AMSTRAD MAGAZINE est une publication strictement indépendante et n'a aucun lien vis-à-vis de la société Amstrad.



PSS et ERE signent un accord



De gauche à droite : Gary Mays (PSS), Emmanuel Viau (ERE), Marc-André Rampon (ERE), Richard Cockayne (PSS).

L'éditeur français ERE informatique et l'éditeur britannique PSS, ont signé un accord de diffusion mutuelle de leurs créations. En fait le contrat va beaucoup plus loin, puisque chaque société, assurera outre la diffusion des logiciels du partenaire dans son pays, la traduction, la fabrication, le support commercial et publicitaire, ceci à partir d'une épreuve "master".

PSS qui avait de gros problèmes de distribution dans l'hexagone, a préféré contacter une société homologue, plutôt que de repartir à la recherche d'un nouveau distributeur. De son côté ERE essayait de pénétrer le marché anglo-saxon. C'est très certainement grâce à la qualité de ses produits que la société pariisiennne a pu conclure cet accord. De plus, les deux éditeurs ont des structures équivalentes et travaillent avec les mêmes objectifs: fournir du logiciel de qualité. Les produits seront distribués sous la marque PSS/ERE.

Les deux éditeurs ont quelques "best" à leurs catalogues : les Flics, Théâtre Europe pour PSS, Macadam Bumper, Mission Delta pour ERE, et ce n'est que le strict minimum, car il en existe pleins d'autres.

Un 6128 à Paris



Heureux parisiens qui n'avaient pas quitté la capitale pour les plages ensoleillées. Ils pouvaient dès le 16 août, certainement en exclusivité, admirer le nouvel ordinateur Amstrad : le CPC 6128. La "bête" était présentée chez "Hyper-CB", une boutique bien connue de tous les amateurs de la marque, où l'on se bousculait pour toucher "avec les yeux", poser des questions et, pour beaucoup, passer une commande. En effet, le 6128 sera disponible chez ce revendeur, dès la dernière semaine d'août pour un prix de : 4490 F en version monochrome et de 5990 F pour la couleur.

La société Amstrad France étant en congés, Hyper-CB nous laissa aimablement prendre contact avec le petit dernier. La

Suite page 9

KARLS TREASURE HUNT



Editeur : Supersavers
Distributeur : Innelec
Support : cassette
Genre : action/réflexe
*Graphisme : ****
*Intérêt : ****
*Difficulté : *****
*Appréciation : ****
Prix indicatif : 120 F

Karl doit retrouver dans une série de pièces, quarante au total, les quarante clés correspondantes. Mais il doit affronter un tas d'obstacles qui se trouvent sur son passage. Toutes les pièces ont leurs lots d'embûches. Mais Karl n'a pas "froid aux yeux" et pour gagner le premier prix, il ferait n'importe quoi. En fait, c'est vous qui n'avez pas peur, car vous devez guider le bon Karl dans cette aventure. Si vous êtes en bonne condition physique et nerveuse, vous mènerez à bien cette aventure, sinon "bonjour les dégâts". Le graphisme du jeu n'est pas dans les plus réussi, on peut faire beaucoup mieux, surtout avec un CPC. Néanmoins on passe un excellent moment en compagnie du bon "vieux Karl".

L'APPRENTI SORCIER



Editeur : Amsoft
Distributeur : Amstrad
Support : cassette
Genre : jeu d'aventure
*Graphisme : ****
*Intérêt : ****
*Difficulté : ******
*Appréciation : ****
Prix indicatif : 99 F

Comme le rêve de votre vie est de devenir sorcier, vous partez à l'aventure pour satisfaire aux épreuves imposées par les grands mages. En effet, ne peut être déclaré sorcier celui qui n'aura pas fait renaître le grand oiseau mystique, d'or et de feu : le "Phoenix". Ce dernier renait de son œuf dans les flammes. Or cet œuf a mystérieusement disparu. Vous devez donc partir à sa recherche, traverser le pays, enquêter auprès des personnages de rencontre. Rien ne doit être laissé au hasard. Pour vous aider tout au long de votre périple, vous pourrez consulter l'indicateur qui se trouve à votre droite, face à l'écran. Il vous renseignera sur les objets que vous portez, ceux que vous trouverez dans les endroits que vous visitez, les possibilités d'orientations offertes. Le jeu n'est pas trop mauvais dans son ensemble, même si l'auteur a oublié la possibilité d'utiliser les directions en abrégé, et d'indiquer sur la présentation la liste des principaux mots-clés. Par exemple, pour interroger un personnage, il faut écrire : parler. Par contre, un gros reproche pour les graphismes utilisant des couleurs trop claires fatiguant rapidement les yeux.

BEACH-HEAD

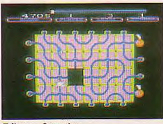


Editeur : U.S. Gold
Distributeur : Run Informatique
Support : cassette
Genre : action/stratégie
*Graphisme : ******
*Intérêt : *****
*Difficulté : ******
*Appréciation : ******
Prix indicatif : 120 F

Beach-Head est très certainement l'un des meilleurs et des plus beaux jeux, que nous ayons eus en essai. D'abord pour son excellent graphisme, ensuite pour ses qualités.

Dans ce jeu, vous devrez vous battre sur la mer, avant de pouvoir lancer vos troupes à l'attaque de la forteresse ennemie, Kuhn-Lin, située sur une île. Tient, ça me rappelle les Falklands ! Beach-Head est divisé en six périodes différentes. La première est la reconnaissance aérienne, qui vous permet ensuite de décider entre deux formes d'attaque : engager vos bateaux dans le passage en profitant de la surprise causée à l'ennemi, ou attaquer ce dernier directement. Ensuite il faut traverser un étroit passage que vos adversaires n'ont pas oublié de protéger par un système de torpillage automatique. Après ces dures épreuves, vers trois heures du matin, l'ennemi lance contre vos bateaux une attaque aérienne. Vous pourrez ajuster le tir afin d'assurer une réelle défense de votre flotte. Malgré les pertes que vous avez fait subir à votre adversaire, celui-ci lance contre vous son armada. Vous pourrez à nouveau ajuster votre tir en utilisant votre joystick ou le clavier. Si vous passez sans trop de dommages ces dures épreuves, vous débarquerez alors deux chars qui affronteront mines, chars ennemis, bunkers, projectiles anti-chars. Vous aurez enfin atteint Kuhn-Lin, il ne restera qu'à la détruire. Un jeu fantastique à ne pas manquer !

CONFUZION



Editeur : Incentive
Distributeur : Run Informatique
Support : cassette
Genre : réflexe/stratégie
*Graphisme : ****
*Intérêt : ****
*Difficulté : *****
*Appréciation :****
Prix indicatif : 85 F

Ce logiciel est à classer dans la catégorie divertissement. Il ne faut pas en attendre beaucoup, sauf de passer un bon moment.

Des empilages de plaques, de forme géométrique carrée ou rectangulaire (dont une est manquante), permettent de faire circuler un mèche. Cette dernière allume, si elle est bien dirigée, une bombe. Une mèche qui se trouve en haut de l'écran sert de chronomètre. Passé le temps imparti, vous aurez bien entendu perdu. Il faudra déplacer les plaques avec adresse et rapidité, ce qui semble très facile, mais qui dans la pratique demande des réflexes, de la stratégie, un calme "olympien". Si vous êtes un grand nerveux, "Confuzion" n'est pas fait pour vous. C'est toutefois un jeu très drôle, qui divertira petits et grands.

LE MYSTERE DE KIKEKANKOI



Editeur : Loriciels
Distributeur : Loriciels
Support : Cassette
Genre : Aventure
*Graphisme : ****
*Intérêt : *****
*Difficulté : ******
*Appréciation : ****
Prix indicatif : 160 F

Le mystère du Kikekankoi est un titre assez ancien au catalogue de Loriciels, puisqu'à l'origine ce jeu avait été développé sur Oric. La nouvelle réalisation pour Amstrad est assez réussie, grâce notamment aux graphismes qui ont été modifiés. L'aventure se déroule dans une grotte. Après qu'une jeune femme vous a envoyé un message de détresse, pour vous avertir qu'elle était retenue prisonnière, vous avez parcouru cinq cent kilomètres pour lui porter secours. Devant la grotte, après quelques instants d'hésitations, vous choisissez votre première direction. A partir de ce moment, que l'éternel puisse vous venir en aide, car rien ne vous sera épargné. Méfiez-vous de tous les objets que vous rencontrerez, certains vous seront utiles, d'autres fatals.

De toute façons, il vous faudra prendre une décision, qui souvent vous renverra au début du jeu. Les chemins empruntés seront aussi révélateurs sur votre chance du moment. Encore un conseil, attention à ce que vous mangez. Au bout de la route, la belle vous attend. Cela ne vaut-il pas quelques risques ? Noter toutes vos actions, cela vous sera très utile en cas d'échec aux premières tentatives.

LA VILLE INFERNALE

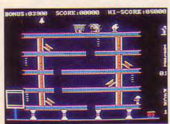


Editeur : Cobra soft
Distributeur : ARG informatique
Support : cassette
Genre : action/stratégie
*Graphisme : ***
*Intérêt : *****
*Difficulté : ******
*Appréciation : ******
Prix indicatif : 120 F

Cobra édite d'excellents jeux, mais a le grave défaut de ne pas soigner leurs graphismes. C'est le cas pour "La ville infernale" qui se révèle comme un superbe jeu, mais avec des dessins absolument nuls.

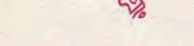
Au volant de votre voiture, vous devez sortir d'une ville qui va rapidement devenir maudite pour vous, si vous n'avez aucun sens de l'orientation. Il n'existe qu'une sortie dans la ville, mais pour l'atteindre, bon courage. Tout ce qu'un automobiliste pourrait avoir comme malchance sur une vie de pilotage, vous arrive en moins de temps qu'il ne faut pour reprendre sa respiration. Si ce ne sont pas les charmants gendarmes qui effectuent un (très) sévère contrôle de votre véhicule, vous avez des chances d'être détroussé par des bandits ; à moins que ce ne soit une petite crevaison qui vous retarde ; ou au mieux que vous ne tombiez en panne d'essence ; mais vous n'aurez très certainement plus d'argent depuis déjà un long moment. Pour vous aider pendant la traversée de la ville, vous trouverez des banques, des garages, des pompistes. Arrêtez-vous dès que possible, car il sera peut-être trop tard ensuite. Surtout que les feux de signalisations ne vous aideront jamais. Un jeu pour muscler son cerveau.

HARD HAT MACK

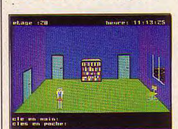


Editeur : Ariola soft
Distributeur : Run Informatique
Support : cassette
Genre : action/réflexe
*Graphisme : ****
*Intérêt : ****
*Difficulté : ******
*Appréciation :****
Prix indicatif : 135 F

La construction d'un building n'est déjà pas chose facile, mais quand en plus vous êtes continuellement gêné, il est certain que le travail n'avancera pas. Le logiciel n'est pas inintéressant, mais souffre d'un côté de "vu et revu". Rien de bien fructueux en fait, sinon que les tableaux ne sont pas aussi faciles qu'on pourrait le penser. Il faut énormément d'énergie et de réflexe pour en venir à bout. Trois tableaux vous attendent. Dans le premier, il faut terminer le travail en mettant en place les portions de poutres manquantes et les riveter, en s'efforçant d'éviter Osha, votre ennemi. Dans le second vous devez récolter des boîtes et vous servir d'un ascenseur de chantier, une poutrelle attachée au filin d'une grue, pour vos déplacements. Attention au vertige. Enfin, dans le troisième, vous devez récupérer des blocs de métal et les apporter à la machine à riveter. Si le jeu n'est pas révolutionnaire, ni l'événement de l'année, on s'amuse beaucoup, avec parfois l'envie de tout casser.



AMELIE MINUIT



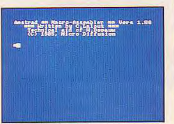
Editeur : Ere informatique
Distributeur : Ere informatique
Support : cassette
Genre : aventure
*Graphisme : ****
*Intérêt : *****
*Difficulté : ******
*Appréciation : ******
Prix indicatif : 140 F

Pendant le chargement du programme, on peut dire que le graphiste de Ere (Michel Rho) nous a gaté en nous offrant une superbe Emilie. Par contre, celle du jeu ne ressemble à rien, sauf peut-être à une tâche de couleur.

Emilie, qui est secrétaire dans la vie, a reçu l'ordre de son patron de retrouver un dossier important. Mais comme personne n'est capable de lui indiquer où se trouve ce dossier, la pauvre secrétaire doit chercher dans les deux cents salles que comporte la tour où elle travaille. C'est pour cela qu'elle a fait appel à vous pour s'en sortir. En prenant l'ascenseur, et en visitant méticuleusement chaque pièces, vous aurez peut-être la chance de mettre la main dessus. Pour vous déplacer, il faut utiliser le pavé directionnel. Dans certain cas, si vous essayez d'ouvrir une porte, on vous répondra que c'est impossible. En réalité, vous devrez chercher une clé dissimulée quelque part dans la pièce. Vous pourrez regarder, prendre, ouvrir, lire, ralentir ou accélérer la marche d'Amélie. Un belle aventure que l'on a plaisir à recharger pour les beaux yeux d'Amélie.



AMS-ASM

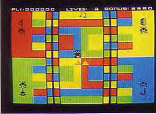


Editeur : Micropuce
Distributeur : Micropuce
Support : cassette
Genre : éditeur macro-assembleur
*Graphisme : **
*Intérêt : *****
*Difficulté : ***
*Appréciation : *****
Prix indicatif : 295 F (335 F disq.)

Dans bien des applications il est souhaitable de travailler en langage assembleur. Micropuce propose pour l'Amstrad son "AMS-ASM", un éditeur macro-assembleur.
L'assembleur est livré avec un manuel de plus de quatre-vingt pages. Ce dernier est loin d'être parfait, mais apporte suffisament d'éléments d'informations pour qui maitrise l'assembleur du Z80. Par contre, dans certaines parties, on peut lire une sorte de "charabia franglais", rapidement énervant. Après avoir rapidement présenté Ams-asm le manuel propose à l'utilisateur de frapper son premier programme démonstration afin de comprendre la structure d'un programme en assembleur. Puis le manuel passe à la description de toutes les instructions utilisables. De nombreux exemples accompagnent chaque instruction.
L'ensemble donnant une approche grandement facilitée pour l'utilisateur. En outre, pour ceux qui connaissent la structure des macro-instructions de l'assembleur IBM 360-370 (gros systèmes), ils retrouveront celles-ci dans Ams-asm. Ce type de macro entièrement paramètrable par l'utilisateur, facilite dans bien des cas l'utilisation d'un programme, en permettant la création d'instructions personnalisées. On aurait été content de trouver avec Als-asm un moniteur-désassembleur, malgré les imperfections du manuel.



JAMMIN



Editeur : Tasket
Distributeur : Innelec
Support : cassette
Genre : action/réflexe
*Graphisme : ***
*Intérêt : ****
*Difficulté : *****
*Appréciation : *****
Prix indicatif : 135 F

Partez à la recherche des instruments de musique. Pour cela empruntez les "ascenseurs" en forme de colonnes de couleurs qui vous transporteront dans les cases où se trouvent les objets convoités. Le logiciel a un graphisme qui fait plutôt penser à un gribouillage d'enfant, mais il est très drôle. La musique qui l'accompagne pa contre est bonne, le tout donnant un logiciel auquel on s'attache vraiment. Le jeu n'est pas facile à terminer, ce qui vous donnera de bons moments à passer avec les "frères rasta". Il se révèle encore meilleur avec en fond sonore un bon vieux tube de Jamaïque. Eviter quand même le "joint", on ne trouve plus le joystick après.

SUPER SONIC



Editeur : Logi'stick
Distributeur : D.D.I.
Support : cassette
Genre : utilitaire sonore
*Graphisme : ***
*Intérêt : *****
*Difficulté : ***
*Appréciation : ******
Prix indicatif : 95 F

Super Sonic est un bon utilitaire sonore, qui permet la création de sons, qu'il est parfois difficile d'obtenir directement avec les instructions du CPC. En effet, beaucoup d'utilisateurs rencontrent quelques problèmes et se découragent rapidement. Avec cet utilitaire, ils ne devraient plus être bloqués. La génération de sons est vraiment très facilitée et n'importe qui est à même d'utiliser le logiciel rapidement. Une assistance est possible pendant toute la durée des opérations. On peut également sauvegarder les sons créés et les rappeler quand on veut. Chaque son est créé grâce à un tableau qui demande de fournir les valeurs du volume, du bruit, du canal, etc. On peut facilement devenir un petit "génie" de la manipulation sonore, comme le fait d'entendre l'exemple donné. Un bon utilitaire bien réalisé et très facile d'emploi. Même s'il n'est pas indispensable pour développer des logiciels, on pourra facilement en tirer profit.



MISSION DELTA



Editeur : Ere informatique
Distributeur : Ere informatique
Support : cassette
Genre : simulateur de vol
*Graphisme : *****
*Intérêt : *****
*Difficulté : *****
*Appréciation : ****
Prix indicatif : 120 F

Comme tous les logiciels de l'éditeur parisien, "Mission Delta" est très bien présenté. La cassette est accompagnée d'un petit livret d'une trentaine de pages, qui vous raconte d'abord une histoire pour vous mettre dans le bain, puis vous trouvez tous les détails concernant le jeu et son utilisation.
Avant de voler seul, il vous faudra passer par la dure école des pilotes. Grâce notamment au manuel parfaitement documenté, vous connaitrez rapidement les secrets et les finesses du pilotage sur votre appareil. Il est vrai que l'on se perd rapidement avec tous les compteurs du tableau de bord : radar, calculateur de bord, compas, horloge, altimètre, variomètre, badin, etc. Sans oublier les nombreuses commandes pour le tir, le parachute, la sélection de l'objectif, les réacteurs, etc. A la sortie de l'école, il est possible de contrôler ses connaissances avec le programme de tests, "Delta T", qui se trouve après le programme principal sur la cassette. Après l'école, vous partirez pour votre première aventure, mais je suis persuadé que vous en vivrez de nombreuses. Si vous reliez l'ordinateur à une chaîne hi-fi, vous obtiendrez un son stéréo des plus réalistes. "Mission Delta" se classe dans le clan restreint de simulateurs, qui simulent réellement le pilotage d'un avion.



Suite de la page 3

première chose que l'on remarque sont les dimensions très réduites de la console, par rapport aux précédents modèles. Une longueur de 51 cm pour 17 cm de largeur et moins de 5 cm d'épaisseur. N'ayez aucune crainte, l'unité de disquette 3 pouces est bien présente. La couleur de la coque est identique à celle du 664. Sur le côté gauche on trouve une prise pour le magnétophone, une autre pour la manette, une sortie son stéréo. A l'arrière, les interfaces pour l'imprimante, le second lecteur et le bus d'extensions ; ainsi que la sortie moniteur plus les deux alimentations 5 V et 12 V.

La disposition du clavier est différente, puisque toutes les touches forment un ensemble et non trois blocs comme auparavant. Seule remarque sur ce clavier visiblement excellent : le "petit enter" ayant disparu, il est remplacé par une touche plus longue, mais disposée dans le prolongement de la barre d'espacement, qui peut poser des problèmes lors des saisies. Le "grand enter" se nomme désormais "return". Le CPC 6128 est livré avec deux disquettes contenant pour l'une le CP/M 2.2, le DR.Logo et Help, pour l'autre le. CP/M Plus et pleins d'utilitaires.

Quelques surprises attendent toutefois l'utilisateur, car sous Basic le 6128 n'offre toujours que 42249 Ko. Il semble que la machine utilise une grande partie de la RAM comme une mémoire vidéo, car les dessins chargés depuis la disquette de démonstration sont rappelés ensuite instantanément à partir de la RAM. Le Basic paraît plus rapide et plus puissant. Sous CP/M, un message nous informe que 61 Ko sont accessibles. Il faut toutefois attendre les résultats d'un essai de la machine, avant d'émettre un verdict quelconque.

Gremlins

Vous vous souvenez des Gremlins, ces charmantes petites créatures capables, à la première contrariété, de se transformer en horribles montres !

A Noël dernier ils ont envahi nos écrans... Et bien maintenant, si vous vous sentez de taille, vous pouvez les affronter sur votre Amstrad.

Gremlins, logiciel de jeu de Aventure International, est disponible chez Run. Vous êtes aux chutes de Kingston en passe d'être envahies par les terribles bestioles. Aidé par votre copain Gizmo vous allez vous débrouiller pour sauver l'endroit du "raz de marée". Vous allez "sauter, examiner, lire, sauver, lancer..." et essayer de ne pas mourir tout le temps !

Amstradivarius Cours de solfège

Techni-Musique sort deux nouveaux logiciels pour les amateurs de belle musique : Amstradivarius et Cours de solfège sur Amstrad CPC 464.

Dans l'ordre, si vous êtes passionné, mais débutant, vous mettrez d'abord en route le cours de solfège et apprendrez les notes en clés de Fa ou de Sol. Pour ce programme, vous devrez possédez une manette de jeu.

Quand le solfège n'aura plus de secret pour vous, vous passerez à Amstradivarius et composerez votre propre partition sur votre micro préféré. Vous écouterez ensuite vos petites musiques de chambre, tout seul si vous le souhaitez, avec lui !

Techni-Musique, rue Fontaine du Bac, 63000 Clermont-Ferrand.

AMSTRADIVARIUS
OMPOSITION MUSICALE 3 VOIES
POUR AMSTRAD CPC 464



D.A.O.-Supercopy

Cobra Soft édite deux nouveaux utilitaires : D.A.O. et Supercopy. Le premier vous permet de créer des dessins sur l'écran de votre Amstrad. Très puissant (certaines routines sont écrites en langage machine) il vous offre plusieurs commandes : tracé, coloriage, cercle, déplacement de zones... Les dessins que vous faites sont ensuite utilisables dans d'autres programmes.

Supercopy est écrit à 100 % en langage machine et vous sert à faire de la copie d'écran sur imprimante. Compatible uniquement avec l'imprimante DMP-1 en mode graphique, avec les autres en mode texte ; il présente un "spooler" qui vous donne la possibilité de programmer pendant un listing de programme.

Cobra Soft, 5, rue Monnot, 71100 Chalon-sur-Saône.



"Association pour la promotion du CPC"

Vous tous, lecteurs et adeptes du CPC, vous avez désormais la possibilité de faire partie d'une nouvelle association : "L'Association pour la promotion du CPC". Le principal lien qui unit les adhérents pour l'instant est une lettre bimensuelle d'information, "La lettre du CPC", vendue uniquement par abonnements.

La lettre et son rédacteur en chef, Patrick Marcelli, se proposent de rendre compte le plus rapidement possible de toutes les nouveautés concernant le monde du CPC. Vous trouverez entre autres informations, des présentations de logiciels, de matériels, des bidouilles ou plus, des articles de réflexion sur des thèmes ciblés.

Outre la rédaction de ce bulletin, "L'Association pour la promotion du CPC" veut mettre en place de nombreuses activités : édition d'un annuaire des utilisateurs, d'un guide des logiciels, mise en place d'une banque de données réservées au CPC et au téléchargement.

Association pour la promotion du CPC, 109, rue Gaston Lauriau, 93100 Montreuil.
Abonnement annuel pour 20 numéros de "La Lettre" : tarif préférentiel jusqu'au 30 septembre, de 120 F

EASY... A VOTRE SERVICE

Tous les "Easy" ont un air de famille bien sympathique ce qui rend très commodes leurs utilisations successives. Gestion du curseur, accès aux options des menus (sympas, les petites fenêtres) se réalisent simplement par les touches du pavé directionnel et vous pouvez choisir le mode écran (couleurs) ainsi que la vitesse de chargement des fichiers. Chaque cassette s'accompagnant d'une documentation où l'essentiel est dit, vous ne regretterez presque plus l'absence de message d'autant que vous pourrez accéder par la touche "COPY" à des mesures d'assistance salvatrices qui s'afficheront en bas de chaque écran (quittez cette aide par la même touche "COPY".

EASY REPORT

Mais le "Must" de cette famille est la possibilité de transposition des fichiers d'un logiciel à l'autre. Vous pourrez par exemple, à partir d'un fichier créé par Easy file, effectuer des calculs complémentaires grâce à Easy calc et obtenir une représentation synoptique de vos résultats par l'intermédiaire de Easy graph. Intéressant, non ? Vous êtes (presque) en possession d'un intégré modulaire... Démystifions : Easy report, qui vous permet (pas toujours sans problème) de réaliser ces transpositions, va vous demander un travail de manipulation important : charger Easy report... charger le fichier données... sauvegarder le fichier 'report'... charger Easy calc... charger le fichier 'report'... traiter... sauvegarder le fichier 'report' traité... etc. etc... Galère! Mais ces opérations nécessitant plus de temps que de matière grise, vous saurez utiliser ce break pour méditer sur des applications que seule votre imagination limitera...

EASY FILE

Ce logiciel laisse à l'utilisateur la liberté d'élaborer lui-même la structure de ses fichiers (1 à 30 rubriques) à partir d'un masque de saisie qu'il créera selon ses besoins et qu'il pourra éventuellement réutiliser grâce à une option de sauvegarde. Chaque rubrique est caractérisée par son identificateur, son type (alphabétique, numérique ou date) et sa longueur (jusqu'à 60 caractères pour les alpha.). Le sous-menu d'utilisation permet de créer les fiches, leur nombre maximun s'affichant en permanence (environ 150 pour une douzaine de rubriques). Le tri (croissant ou décroissant) pourra se faire par ordre alphabétique, numérique ou chronologique en fonction du type de la rubrique. Les données numériques peuvent être saisies ou calculées selon une formule définie lors de la création de l'écran, et totalisées pour l'ensemble du fichier. Si les mises à jour (adjonction, suppression ou modification d'articles) et la consultation se réalisent aisement par l'option "recherche d'une fiche", la sélection d'un enregistrement semble être le point faible de ce logiciel. Elle s'effectue soit sur une caractéristique, soit de manière cumulative si on en choisit plusieurs, et ne peut se réaliser qu'en deux étapes (format à définir puis recherche). Gênant... Tout autant que ce "Bug" (non dévastateur, je vous rassure) que vous ne manquerez pas de constater au niveau des dates. Vous pourrez éditer tout ce qui concerne la structure du fichier (masque de saisie, format...), recopier la fiche dont vous avez sélectionné l'affichage et réaliser l'impression partielle ou totale du fichier avec des options intéressantes (titre, interligne, commentaire...). Les limites imposées à la gestion de fichiers par l'accès séquentiel ont été vérifiées, et bien qu'on ne manquera pas d'être angoissé lors de certains arrêts prolongés de l'ordinateur, Easy File s'avère un outil efficace aux applications variées et ses possibilités de traitement séduiront tout autant le néophyte que le grand initié.

EASY CALC

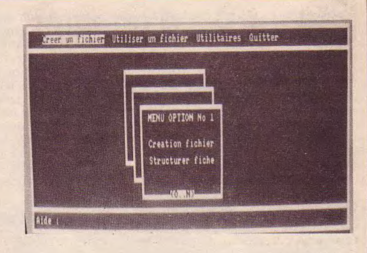
L'option "tableau" du menu principal vous permet de visualiser à la fois 50 cellules (5 colonnes sur 10 lignes) des 780 (26 × 30) que comporte la feuille de calcul. Vous décalerez cette grille initiale par les touches du curseur et promènerez la case de travail de la même façon mais en appuyant à la fois sur la touche 'Shift'. Les huit commandes du calc s'obtiennent en ne frappant que la première lettre de leur nom: ce qui vous familiarisera très vite avec ces fonctions :

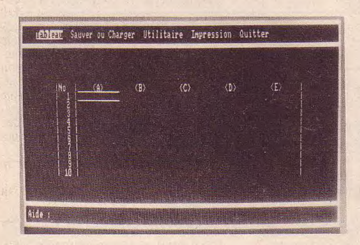
A(ffiche) le contenu de la case sur la première ligne de l'écran.
E(tiquette) : entrée d'une donnée alpha ou numérique.
F(ormule): entrée d'une formule de calcul
D(éplacement) aux coordonnées précisées (Al par défaut) de la case de travail qui s'affiche en haut à gauche de la grille.
S(ystème): définit le format des données numériques (entier, flottant ou scientifique) à afficher ou éditer.
C(alcule) et affiche les résultats (mise à jour de la feuille).
T(ransposition) des étiquettes ou des formules de la case de travail vers la ou les cases définies par leurs coordonnées et le pas de transposition.
Q(uitter): retour au menu prinicipal.

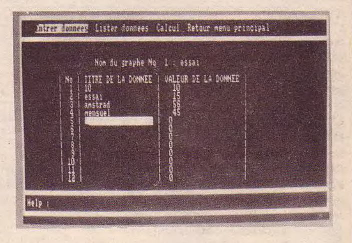
Cette commodité d'utilisation est, hélas, entachée de l'obligation de préciser à chaque fois la fonction utilisée. Gênant lorsque l'on doit, par exemple, entrer plusieurs données successivement. Mais le talon d'Achille de calc réside, semble-t-il, dans l'absence de fonction assurant des calculs essentiels. Il est possible de pallier cette carence par des formules à rallonge ou des cogitations épuisantes. Le menu d'impression vous propose d'éditer soit les formules, soit les étiquettes, soit une partie de la feuille de calcul que vous définirez par les coordonnées des cellules extrêmes. Quant un tableur n'est pas professionnel, on ne peut s'attendre à le voir nous inviter à











décrocher la lune : alors, restons sur terre et utilisons Easy calc pour nous initier aux feuilles de calcul et/ou pour réaliser des applications limitées mais honnêtes.

EASY GRAPH

Compagnon apprécié d'Easy calc, ce logiciel graphique permet le tracé de figures et effectue également des calculs statistiques (moyennes, écart-type, variance...). Sur un graphe, vous pourrez représenter un maximun de douze données (titre valeur), saisies au clavier ou chargées à partir d'un fichier déjà constitué. Chaque graphe est identifié par un numéro (1 à 40 maximun) et par un titre. L'accès aux différents modes de tracés s'effectue par manipulations rapides et faciles à réaliser. Easy graph fonctionne en mode haute résolution, 640 × 200 pixels, en deux couleurs. Vous choisirez l'histogramme bâtons à deux ou trois dimensions, le graphe à lignes ou, par tradition, le camembert (seule cette dernière option vous affiche les valeurs des données représentées). L'échelle n'est pas des plus pratiques à lire par contre son calcul permet d'occuper toute la place disponible. L'impression du graphe affiché pourra se réaliser après avoir choisi le type d'imprimante (compatible EPSON ou AMSTRAD DMPI) et le mode (normal ou inversé).

Report	File
Calc	Graph
	Bank

Entrer une donnee Charger un fichier Quitter
Operation numero : 1
Maximum d'operations : 527
Date de l'operation :
(C)rediteur /(D)ebiteur :
Montant de l'operation :
(C)heque /(A)utre/(E)speces :
Ordre de l'operation :
Aide :

Malgré ses insuffisances au niveau du paramètrage du graphique, Easy Graph s'installe dans la gamme des logiciels simples à utiliser, sans prétentions mais offrant néanmoins la possibilité de réaliser des applications exploitables. Alors... faites donc "causer vos chiffres"!

EASY BANK

Voici un produit qui s'adresse aux fanas des carnets de chèques et des cartes de crédit, aux angoissés de la lettre d'injonction, aux inconditionnels de la gestion par micro... (rayer la ou les



mentions inutiles). On sait que la tenue d'un compte bancaire sur ordinateur a ses adeptes et ses refoulés. Alors si ça vous branche, sachez que vous pourrez périodiquement introduire 527 opérations (débitrices ou créditrices) et suivre ainsi les fluctuations du solde de votre compte. Les entrées (ou sorties ??) se font à partir d'un masque de saisie complet et bien dessiné, mais non modifiable... Il y est prévu, en particulier, le mode d'agression ou d'alimentation de votre avoir: C pour chèque, E pour Espèces et A pour Autre (à définir éventuellement). On aurait apprécié un éventail plus large : carte de crédit, prélèvement automatique, virement... Ces données étant en place, libre à vous de les faire défiler, de les modifier, de les annuler ou de les rechercher selon un ou plusieurs critères. Ne vous privez pas de demander l'édition et de la classer dans vos archives : elle fait très balance de pro. Enfin et pour le plaisir, consultez une dernière fois le solde de votre compte qui s'affichera, bien cadré sur votre écran (là, on vous gâte) et n'oubliez pas, bien sûr, de sauvegarder toutes ces précieuses informations.
Voilà les limites des performances d'Easy bank. N'espérez pas dresser des états de rapprochement avec votre extrait de compte, ni trier vos opérations chronologiquement ou par libellé, ni insérer des mouvements oubliés... Easy bank fait ce qu'il peut et même s'il le fait bien, on lui reprochera quand même d'être un peu "short"...

Michel Lepourcelet

Nom du logiciel	**Easy Report**	**Easy File**	**Easy Calc**	**Easy Graph**	**Easy Bank**
Créateur	POWER SOFT	POWER SOFT	POWER SOFT	POWER SOFT	POWER SOFT
Distributeur	POWER SOFT	POWER SOFT	POWER SOFT	POWER SOFT	POWER SOFT
Prix public	120 F	180 F	180 F	180 F	180 F
Format	cassette	cassette	cassette	cassette	cassette
Genre	utilitaire	fichiers	tableur	graphique	gest. familiale
Graphisme	★	★★	★★	★★	★★★
Intérêt	★★★★	★★★★	★★★	★★★	★
Difficulté	★★	★★	★★★	★	★
Appréciation	★★★	★★★★	★★★	★★★	★★

MEA CULPA

Dans le premier numéro d'Amstrad Magazine, vous avez constaté les nombreuses erreurs qui se trouvaient dans certains listings. Veuillez accepter toutes nos excuses pour ce dérangement et trouver ci-dessous les corrections à apporter :

Page 11, tous les 'S' de NOMS, ADRS, TELS, ENCORES, ENREGEGS, ES, sont à remplacer par '$' (Dollars). Les '£' (Livres) sont à remplacer par '='.

Page 41 et suivantes (Mission pour pachebran), modifier les lignes :

```
-550- enlever le ':' qui suit le 'Print'.

550 PRINT "JE NE PEUX PRENDRE CA" : GOTO 210

-860- il manque ':'.

860 GOSUB 20 : IF z9=2 THEN 210

-890- il manque 'GOTO 210'.

890 IF R=6 THEN PRINT "CA PARAIT LOURD" : GOTO 210

-980- il manque ':'.

980 IF R=41 AND BU=1 THEN PRINT "L'ECRAN EST NOIR" : GOTO 210
```

Si vous éprouvez quelques difficultés pour commencer le jeu, donnez l'ordre : déposer badge ; puis ensuite : entrer bâtiment. Vous serez alors à l'intérieur du bâtiment. A vous de jouer !



TURBO PASCAL sur AMSTRAD

**La Société FRACIEL* vient d'adapter la version 2.0 du Turbo Pascal pour Amstrad.
Ni une nouvelle machine, ni un système d'exploitation encore inconnu, ni un langage inédit, TURBO Pascal est un "simple" compilateur. C'est-à-dire un programme qui prend du texte en entrée et en ressort une suite d'instructions exécutables par la machine.**

Evidemment, le texte de départ n'a rien à voir avec du L.F. Céline ! Il s'agit du langage de programmation PASCAL mis au point, il y a une quinzaine d'années par Nicklaus Wirth (professeur à l'Université technique de Zurich) dans le but d'apporter clarté et rigueur - en un mot : structure - à la science nouvelle de la programmation. Si BASIC en effet, est d'un accès facile aux néophytes, il devient vite trop embrouillé pour un programmeur plus assidu. Et comme l'état de néophyte n'est que provisoire, tout amateur est vite tenté par l'expérience PASCAL.

Seulement, il reculera peut-être devant les moyens de mise en œuvre du langage compilé du professeur Wirth. BASIC, pour sa part, utilisant un interpréteur, plutôt qu'un compilateur, est immédiatement (ou presque) éxécutable. Et surtout, ce mode de fonctionnement permet une grande souplesse d'utilisation, la même structure permettant l'édition puis l'éxécution des programmes, ainsi que leur mise au point pas à pas. Avec Pascal, de telles facilités sont exclues. La richesse même de la structure (déclaration des variables, des types, déclaration des fonctions et procédures) réclame que le programme soit complet avant que de pouvoir être confié au compilateur. Ce qui signifie que les travaux d'édition, de compilation et enfin d'éxécution doivent être assignés à des modules séparés. Et l'utilisateur doit saisir son programme, le sauvegarder, sortir de l'éditeur pour appeler le compilateur, attendre la fin de la compilation pour pouvoir enfin lancer l'éxécution ! Mais on arrive rarement au bout de cette suite d'opérations du premier coup, car il manque toujours un point virgule ou un "end" et il faut alors revenir sous l'éditeur afin de corriger l'erreur pour ensuite refaire appel au compilateur...

Que choisir, un langage simple et bien maniable même sur une installation rudimentaire, comme BASIC, mais qui s'avère très vite inadéquate à une programmation sophistiquée ou un langage plus structuré, donc mieux adapté aux problèmes complexes mais qui demande une manipulation plus pesante avant sa mise en marche par la machine ?

Une extraordinaire rapidité de compilation

Il semble que TURBO Pascal soït arrivé à point nommé pour résoudre le dilemme. TURBO Pascal est un simple compilateur, mais un compilateur turbo ! C'est là sa première caractéristique : une rapidité de compilation telle que sur des programmes relativement simples on a l'impression d'avoir affaire à un langage interactif. Cette impression est de surcroît renforcée par le fait que la faible place qu'occupe TURBO Pascal en mémoire (28K pour les CPU 8 bits et 36 K pour les 16 bits) a permis de lui adjoindre un éditeur de texte, dont les commandes obtenues en couplant la touche CTRL avec un caractère alphabétique s'apparentent à celles de Wordstar. Ce confort, déjà appréciable pour un langage compilé est agréablement agrémenté par des détails de programmations dus à Philippe KHAN, créateur et fabricant du TURBO Pascal, tels que le retour direct

Suite page 16



L'unité de disquettes Amstrad DDI-1

AMSTRAD commercialise une unité de disquettes munie d'une interface permettant la connexion au CPC 464.

Vendue 2 490 F avec DR. Logo et surtout le système d'exploitation CP/M 2.2, cette extension semble destinée à autoriser l'orientation du CPC 464 vers des applications professionnelles.



L'emballage qui vous est remis lorsque vous parvenez à vous procurer le précieux matériel contient : protégé par un boîtier de polystirène, le lecteur de disquettes ; l'interface avec l'unité centrale et son câble de raccordement ; une disquette 3 pouces contenant CP/M 2.2 et DR. Logo, un manuel d'utilisateur.

Au premier abord le lecteur de disquettes donne, malgré sa petite taille une impression de robustesse qui sera confirmée à l'utilisation. Les manipulations de disquettes effectuées, bien qu'étant nombreuses et faites sans précautions particulières, n'affecteront pas son fonctionnement.

Il n'en est pas de même pour l'interface, qui semble un peu légère. Constituée d'un unique boîtier de plastique destiné à être connecté à l'arrière de l'unité centrale, elle apparaît comme le maillon faible, de l'ensemble, en particulier, au niveau de la fiabilité de la broche de connexion. A l'usage d'ailleurs, et bien que les instructions de branchement et de mise en route du drive aient été respectées à la lettre, il se révèlera souvent nécessaire d'arrêter puis remettre en route l'ordinateur, le système se refusant obstinément à reconnaître la présence de l'unité de disquettes.

Le manuel utilisateur est de la même facture que celui du CPC 464, c'est-à-dire bien réalisé et, complet. Il indique clairement les instructions supplémentaires apportées au Basic, ainsi que le mode opératoire des principaux utilitaires de CP/M. Un chapitre entier de ce manuel est destiné à DR. Logo. On peut toutefois regretter que ses réalisateurs n'aient pas jugé bon de donner des indications complémentaires sur les utilitaires fournis avec CP/M, et en particulier sur l'utilisation de l'assembleur 8080.

Les possibilités réelles du drive DDI-1

Si le drive DDI-1 offre effectivement tout ce que promettent ses annonces, il ne permet pas de tirer tout le parti possible de l'adjonction d'un support disquette. Les problèmes posés sont de plusieurs ordres.

Problèmes liés au matériel

Ces problèmes étaient prévisibles dès l'annonce du matériel, et son liés au format de disquette choisi. En effet, le format de disquette 3 pouces, est peu répandu et le nombre de fournisseurs étant très limité, les pénuries de disquettes ont été fréquentes (d'aucuns doivent se rappeler, leur rareté au début du mois de juin 1985). Ces difficultés ne devraient être pour ceux qui les ont vécues que de mauvais souvenirs. AMSTRAD aurait acheté à HITACHI ses unités de fabrication de disquettes 3 pouces, et serait décidé à assurer un approvisionnement régulier du marché. On en a pour preuve le fait que, depuis le début du mois de juillet, les disquettes trois pouces de marque AMSTRAD ou SCHNEIDER se trouvent en quantités conséquentes chez les revendeurs.

Le format est d'un prix élevé (minimun relevé à ce jour sur Paris, 590 F les onze disquettes). Sa robustesse, liée à son enveloppe plastique, ne saurait faire oublier ce fait.

Il n'a, sur le drive AMSTRAD qu'une capacité de 160 Ko formattés par face. Avec deux lecteurs, on n'atteindra que 320 Ko, soit une capacité inférieure à celle d'un seul lecteur de l'IBM-PC avec lequel, ou moins pour le format des disquettes, le DDI-1 prétend être compatible sous CP/M.

Notons toutefois que le DDI-1 FIRMWARE d'AMSTRAD (réf. SOFT 158A) précise que, moyennant une modification de leurs connexions, il est possible d'utiliser certains drives de 5,25 pouces, tant en première qu'en deuxième unité.

Problèmes liés au logiciel

LE BASIC

Le Basic AMSTRAD gère les fichiers sur disquettes comme les fichiers sur cassette. Il n'existe pas de possibilité d'accès direct sur disquette et, de l'aveu même d'AMSTRAD France, un Basic ayant ces possibilités n'est pas prévu actuellement.

En outre les instructions MERGE et CHAIN MERGE provoquent le message d'erreur "EOF MET". Vous n'éviterez cet inconvénient qu'en sauvegardant vos programmes sous forme de fichiers ASCII, ou en utilisant une routine spécifique à inclure dans vos programmes Basic.

Enfin, la place disponible en mémoire diminue légèrement et est de 42249 octets. Ceci risque de vous poser des problèmes pour les très gros programmes.

CP/M

Compte tenu de ce qui est écrit au paragraphe précédent, les fonctionnalités de

CP/M ne peuvent pas être utilisées par le Basic. Vous ne pourrez profiter pleinement des possibilités de ce système d'exploitation qu'en utilisant l'assembleur et en ayant la brochure DDI-1 FIRMWARE, vendue 245 F.

Il faut donc considérer que CP/M est fourni pour permettre la mise en œuvre de logiciels professionnels sur le CPC 464. Or une telle éventualité dépendra du bon vouloir des éditeurs de logiciels et on ne peut pas dire qu'il y ait actuellement "du monde au portillon". De plus, les rares logiciels prévus sont chers par rapport au prix du matériel (environ 4 500 F annoncés pour l'ensemble Wordstar-Mailmerge, pas encore prêt pour des raisons d'occupation mémoire).

Conclusion

Pour un prix sans concurrence sur le marché actuel, AMSTRAD propose une unité de disquettes qui, si elle ne permet pas de transformer immédiatement le CPC 464 en ordinateur professionnel, offre quand même des perspectives intéressantes.

Si les éditeurs se décident à adapter leurs logiciels fonctionnant sous CP/M et si, d'autres part, AMSTRAD publie un Basic gérant l'accès direct sur disquette ainsi que (ça peut être nécessaire) une extension mémoire, cette unité deviendra autre chose qu'un simple moyen de lire et sauvegarder les programmes ou qu'une fantaisie de hobbyiste. Le matériel AMSTRAD, en proposant pour 5480 F une configuration avec moniteur monochrome, 42 Ko de mémoire utilisateur, et une unité de disquettes, offrirait un rapport utilisation/prix exceptionnel.

R.P. Spiegel

Comment utiliser Merge et Chain Merge avec le drive

La présence d'une unité de disquette empêche sur le CPC 464, d'utiliser les fonctions Merge et Chain Merge. La routine suivante vous permettra de mettre en œuvre ces fonctions sans avoir le message "EOF met".

```
100 MEMORY HIMEM-41
105 h=HIMEM
110 DEF FNf1(a)=&FF AND INT
    (a/256)
115 DEF FNf2(a)=&ff AND UNT(a)
120 FOR i=h+1 to h+38
125 READ b: POKE i,b
130 NEXT i
135 POKE h+3,f2(h+39)
140 POKE h+4,f1(h+39)
145 POKE h+9,f2(h+41)
150 POKE h+10,f1(h+41)
155 POKE h+18,f2(h+1)
160 POKE h+19,f1(h+1)
165 POKE FOR i=0 to 2
170 POKE h+39+i,PEEK(&BC80+i)
175 NEXT
180 POKE &BC80,&c3
185 POKE &BC81,f2(h+1)
190 POKE &BC82,f1(h+1)
200 DATA &e5,&ea,0,0,&22,&80,&bc
205 DATA &3a,0,0,&32,&82,&bc
210 DATA &cd,&80,&bc,&21,0,0
215 DATA &22,&81,&bc,&21,&80,&bc
220 DATA &36,&c3,&e1,&d8,
    &c8,&fe,&1a
225 DATA &37,&3f,&c0,&b7,&37,&c9
```

Vous pourrez soit :
- éxécuter cette routine par RUN, auquel cas vous pourrez ensuite reprendre vos programmes par MERGE.
- inclure cette routine dans ceux de vos programmes utilisant la fonction Chain Merge.

Suite de la page 14

TURBO PASCAL sur AMSTRAD

sous éditeur à la ligne où le compilateur détecte une erreur : finis les affres des points virgules oubliés !

Dans ces conditions, TURBO Pascal se présente comme un bon "interface" entre l'utilisation interactive du BASIC et la programmation plus rigide des langages compilés. Les utilisateurs de petits systèmes pourront ainsi abandonner sans trop de regrets les labyrinthes des numéros de lignes que génère habituellement leur BASIC pour les contrées rendues plus faciles d'accès des couples "begin:end" et des appels récursifs.

Mais que l'on ne se méprenne pas: le PASCAL du TURBO est, à quelques variantes près, fidèle au langage défini par N. Wirth et K. Jensen en 1971. L'amélioration apportée par Philippe Khan porte essentiellement sur une optimisation des CP/M 80 (et 86), PC-DOS et MS-DOS quant à l'utilisation du langage. Ce n'est que dans la version 2 du TURBO que l'on retrouve l'instruction "dispose" permettant de libérer les pointeurs que l'on n'aura plus à utiliser. En revanche, cette version introduit la possibilité de recouvrement (en anglais : overlay). C'est à dire qu'une partie du programme (fonctions et procédures) peut n'être chargée, en cours d'exécution, qu'au moment où le programme principal en a besoin. Ce qui permet de faire tourner un programme qui ne tient pas tout entier dans l'espace mémoire par appel sélectif à telle ou telle partie réclamée par l'exécution.

Ces parties externes au programme peuvent tout aussi bien être des procédures figurant dans un fichier, que des options de compilation portant sur la non-déclaration du type de certains paramètres et l'édition de liens ainsi que la possibilité de leur déclaration par anticipation (c'est-à-dire : sans qu'elles figurent dans le texte du programme) rendues intégrables au corps du programme. A vous de constituer alors votre bibliothèque de sous-programmes.

Utilitaires en fichier source

En attendant, vous pourrez toujours utiliser la TOOL-BOX que vous propose P. Khan et sa société, BORLAND International. Il s'agit de trois utilitaires (que vous pourrez décortiquer à loisir puisqu'ils sont présentés en fichier source) dont le principal est un logiciel de gestion de fichier : TURBO Access. Il permet principalement l'accès dit "aléatoire" aux enregistrements d'un fichier. L'accès aléatoire se caractérise par l'accès direct à un enregistrement si l'on en connaît le numéro. Pour connaître le dit numéro, TURBO Access (et ses fonctions) se sert d'un fichier d'index contenant les clés, ou indentificateurs référant aux enregistrements. Dans ces index figurent également les numéros des enregistrements référés. Par exemple: un enregistrement contenant les informations relatives aux transactions avec une société sera indexé par le nom de cette société. Le fichier d'index est structuré en arbre-B (son organisation est relativement bien exposée dans le manuel accompagnant la disquette). Ce type de structure optimise la recherche et fournit rapidement, sans avoir à parcourir tout le fichier, le numéro d'enregistrement de l'information désirée permettant ainsi son accès direct. La perte de place mémoire est alors compensée par la rapidité d'accès. Mais figurent également dans la "boîte à outils" les procédures de création, d'insertion ou suppression d'enregistrement. On trouve aussi dans la même boîte un algorithme de tri assez sophistiqué (tri multi-clés, entre autres).

On peut alors se demander si les productions de BORLAND International sont de simples logiciels, tant leur lien avec le système qui les reçoit paraît modifier les possibilités de ses derniers. Alors, ni nouvelle machine, ni système d'exploitation inédit... mais quoi : machine logique ?

Pascal Manoury

*FRACIEL, 42, rue des Prébendes
37000 TOURS
Prix public de la version 2.0 environ 750 F



Smith-Corona Fastext 80



Une bonne affaire

Toujours disponible sur le marché, l'imprimante Fastext 80 de Smith-Corona est une très bonne affaire, pour ceux qui recherchent une machine de qualité pour un prix très abordable.

Petit prix, mais il fait le maximun

Proposée pour un prix d'environ 2.200 F, la Fastext 80 en donne pour son argent. Cette imprimante à impact par points accepte différents formats de papier: minimum 10,16 cm à 27,94 cm en feuille à feuille, 25,39 cm en listing, 21,59 cm en rouleau. L'avancement du papier est du type friction en version de base, mais une option traction est disponible (nécessaire pour le papier listing). La cassette-ruban a une durée d'environ un million de caractères.

La mécanique est installée dans une coque de plastique de couleur crême claire. Ses dimensions sont 21,5 × 40 × 9 cm sortie de l'emballage, et 30 × 40 × 14 après installation de la grille guide papier et de l'option traction. A l'arrière de la machine, on trouve une entrée 8 bits parallèles de type "centronics" ; l'interrupteur de mise sous tension, qui est contrôlable par un leed rouge disposé à l'avant. A côté de ce dernier est située la commande "on line/off line" assurant les tests et la communication avec l'ordinateur. On ne rencontre aucun problème pour le branchement du périphérique et son utilisation.

Une excellente qualité d'impression

La Fastext 80 offre trois types de caractères différents (normal, concentré, et condensé), plus un mode étiré et un mode souligné, utilisables avec chaque type. Ce qui, en réalité, donne douze possibilités différentes : caractère, caractère souligné, caractère étendu, caractère étendu souligné. On dispose également des caractères accentués français, accessibles après sélection. Il existe aussi un mode graphique normal, 480 points par ligne et un mode dit "table traçante", de 576 points par ligne.

La tête d'impression a huit aiguilles, fonctionne en bi- ou unidirectionnelle. L'impression est réalisée en huit points pour la hauteur, le huitième assurant les jambages et le soulignement, et de cinq à dix points en largeur suivant le mode utilisé. La qualité d'impression de la Fastext 80, sans être exceptionnelle, est d'une qualité très satisfaisante. Même si les jambages ne sont pas véritablement descendants. On obtient de 40 caractères par ligne (mode normal étiré) à 132 caractères (mode condensé).

Le bruit émis par la machine lorsqu'elle travaille n'est pas trop élevé, ce qui n'est pas le cas pour beaucoup d'imprimantes. L'accès aux différents type de caractères et les spécifications de fonctionnement : saut de page, valeur de l'interligne, nombre de lignes par page, impression unidirectionnelle etc, sont indiqués par des ordres précédés de l'instruction "Print # 8" pour l'Amstrad.

Petit prix, mais gros problème

Dans le cas particulier des imprimantes Smith-Corona, il subsiste un doute importants : trouvera-t-on encore des machines dans six mois, le service après-vente sera-t-il assuré ?

En effet, la société Smith-Corona France n'existe plus. La distribution du matériel étant assurée par l'entreprise Métaphax, spécialisée dans la micro-informatique, les matériels de reprographie, les supports magnétiques et les consommables (papier listing, encre pour photocopieurs, rubans, etc). Francis Caminade, chef de produit informatique, nous a précisé que Métaphax dispose encore de machines en stock, et que les réparations éventuelles seraient assurées, car pour le moment c'est l'échange systématique qui est pratique.

Philippe Lamigeon

```
Caractères accentués Français:
é ç à è ù ^

Mode Normal
Mode Normal Etire
Mode Condense
Mode Condense Etire
Mode Concentre
Mode Concentre Etire
Mode Souligne
Mode Souligne Etire
```

A la découverte d'une fonction "oubliée"

En avant première pour AMSTRAD MAGAZINE, voici un extrait du prochain livre de Daniel Martin, "*Le livre de l'Amstrad*". Ce passage traite d'une fonction que les concepteurs de l'AMSTRAD ont "oublié" de mentionner dans le guide de l'utilisateur. Cette fonction bien utilisée peut faire des merveilles ! Alors bidouilleurs, à vos claviers.

La fonction ∂ (VARPTR)

Syntaxe: X = ∂ var ou PRINT ∂ var.
Où var représente une variable quelconque.

La fonction ∂ (VARPTR) est l'un des plus merveilleux outils du BASIC. Elle permet d'atteindre l'adresse de stockage des valeurs assignées aux variables ainsi que différentes informations sur leur contenu. A l'aide des adresses obtenues par la fonction ∂ , de l'instruction POKE et de la fonction PEEK, on peut effectuer une foule d'opérations très utiles.

L'utilisation principale de la fonction ∂ est certainement de retrouver des informations sur les chaînes de caractères. Lorsqu'on écrit : 10 A$ = «COUCOU», le système d'exploitation de l'interpréteur BASIC doit sauvegarder la valeur affectée à A$ (en l'occurence COUCOU) quelque part dans la mémoire. Lorsque, quelques lignes plus bas, on écrit: 50 PRINT A$, le système devra être capable de retrouver COUCOU.

Pour effectuer cette opération, le BASIC possède une liste des variables utilisées. Chaque fois qu'il rencontre une nouvelle variable, il l'ajoute à la liste. La variable qui a été rencontrée la première dans le programme sera la première dans la liste, et celle qui sera rencontrée la dernière dans le programme, sera la dernière dans la liste. Chaque fois que le BASIC rencontre une nouvelle variable, il fouille la liste pour voir si cette variable a déjà été affectée. Si ce n'est pas le cas, il l'ajoute à la liste. Le BASIC possède deux listes: une pour les variables simples, et une pour les variables dimensionnées. Le système consulte la liste appropriée à la variable rencontrée.

Remarque : le temps pris par le système pour retrouver une variable est un facteur influençant très fort la vitesse d'éxécution des programmes. Il est possible d'améliorer de façon notable la vitesse d'éxécution d'un programme en définissant au début du programme une liste des variables les plus souvent utilisées.

Les variables simples sont définies la première fois qu'on leur attribue une valeur, les variables tableau sont définies lors de l'instruction DIM. En plus du nom de la variable, la liste contient des informations sur le type de variable. En fonction du type, d'autres informations sont fournies au système : soit la valeur de la variable, soit l'adresse où l'on peut trouver cette variable.

Le BASIC utilise ces informations pour retrouver rapidement les valeurs des variables lors de l'éxécution d'un programme. Toutes ces informations sont aisées à accéder grâce à la fonction ∂ (VARPTR pour VARIable PoinTeuR).

L'utilisation de ∂

L'instruction X = A$ fournira une valeur X, adresse où des informations sur A$ pourront être trouvées. La variable sur laquelle on demande des renseignements peut être une variable entière, une variable flottante, une variable alphanumérique ou un élément d'une variable tableau de n'importe quel type.

Exemple: X = ∂A$(2) est parfaitement licite.

L'utilisation que l'on peut faire de l'adresse contenue dans X à l'issue de l'instruction est fonction du type de la variable. La valeur contenue dans X étant une adresse, elle est comprise entre 0 et 65535. Donc c'est une valeur entière et elle tient sur 2 octets.

Contenu de AD dans la fonction AD = ∂var

Le contenu de l'adresse fournie par la fonction ∂ varie en fonction du type de variable. Si AD est l'adresse fournie par la fonction ∂ alors quel que soit le type de la variable :
l'adresse AD-1 contient le type de la variable:
1 = Variable entière
2 = Variable alphanumérique
4 = Variable flottante.

L'adresse AD-2 contient le dernier caractère du nom de la variable avec le bit 7 à 1 (augmenté de 128).

De AD-3 à AD-n on trouve les autres caractères du nom de la variable s'ils existent.

Exemple: si ANNE est une variable chaîne qui contient "TOTO" alors:

AD-5	AD-4	AD-3	AD-2	AD-1	
A	N	N	.	.	(ASCII)
41	4E	4E	C5	2	(HEXA)



Le contenu des adresses AD et suivantes, dépend du type de la variable. Mais quelque soit le type de variable, le nombre d'octets à considérer vaut l'indicateur de type (AD-1) augmenté de 1. Ainsi si la variable est entière, il faut considérer deux octets (1 + 1).

Si la variable est:

Une variable entière

Une variable entière est mémorisée sur 16 bits en binaire signé. Le bit le plus significatif (B15) est le bit de signe.

AD contient les 8 bits les plus significatifs de la variable (B0 à B7).

AD + 1 contient les 8 bits les plus significatifs de la variable (B8 à B15).
Exemples :
a) si la variable entière vaut 12345 décimal (3039 en hexa), alors :
l'adresse AD contient 57 (39H) et AD + 1 contient 48 (30H) et 57 + 256 × 48 = 12345

```
Faites :
10 DEFINT A
20 A=12345
30 AD=@A
40 PRINT PEEK(AD);PRINT(AD+1)
50 PRINT PEEK(AD)+256*PEEK(AD+1)
```

b) si la variable vaut -12345 (CFC7H), l'adresse AD contient 199 (C7H) et AD + 1 contient 207 (CFH) et 199 + 256 × 207-65536 = -12345.

Une variable flottante

Les 5 octets de AD à AD + 4 contiennent la valeur de la variable. La façon de représenter une variable en 5 octets est particulièrement complexe à expliquer. Essayons par quelques exemples de comprendre le mécanisme. Les valeurs données pour AD0 à AD4 sont en hexadécimal, la valeur de la variable est en décimal.

var	AD+0	AD+1	AD+2	AD+3	AD+4	
0	0	0	0	0	0	FACILE
1	0	0	0	0	81	DEJA PLUS COMPLEXE
2	0	0	0	0	82	COMPRIS ?
8	0	0	0	0	84	OUI ?
-8	0	0	0	80	84	BIZARRE
-16	0	0	0	80	85	....
0.5	0	0	0	0	80	
3	0	0	0	40	82	
2.5	0	0	0	20	82	

Constatation:
- 0 est une exception facile à comprendre. Le bit B7 de AD + 3 indique le signe 0 = positif, 1 = négatif. AD + 4 - 81H représente la puissance de 2 pour obtenir le nombre inférieur le plus proche au nombre cherché. (8 = 2 exposant 3 (84-81)). Autrement dit si le nombre contenu dans var est m alors AD + 4 contient le logarithme en base 2 de m augmenté de 81H.

Les autres bits doivent être lus en partant de B6 de AD + 3 jusqu'à B0 de AD. Ils indiquent la fraction du nombre représenté par AD + 4 qui doit être ajoutée pour obtenir le nombre désiré. B6 de AD + 3 vaut 1/2. B5 vaut 1/4... B0 vaut 1/64 et ainsi de suite avec les bits de AD + 2, AD + 1 et AD.

Exemples:
a) si AD + 4 contient 88H et AD + 3 contient 60H alors le nombre est : 2 exposant 88-81 soit 2 exposant 7 = 128. AD + 3 = 60 = 01100000 c'est-à-dire + 1/2 + 1/4 de 128 autrement dit 128 + 64 + 32 = 224.

b) représentons le nombre -12345
- négatif: B7 de AD + 3 = 1
- LOG base 2 de 12345 = 13 ou OD PRINT LOG (12345)/LOG(2)
- 2 exposant 13 = 8192
- AD + 4 contient donc 81H + ODH = 8EH
- Exprimez le nombre en binaire PRINT BIN$ (12345) 11000000111001
- Mettre à 0 le premier digit significatif. 01000000111001.
- Complétez par des 0 à droite pour obtenir un multiple de 8 bits. Coupez par tranche de 8 bits.
01000000 11100100
- Vous obtenez les valeurs respectives de AD + 3, AD + 2,....
- Mettre le bit de signe de AD + 3 (B7) 11000000 11100100

AD + 3 = CO, AD + 2 = E4, AD + 1 = 0 et AD = 0.

Une variable alphanumérique

AD contient la longueur de la chaîne en nombre d'octets. AD + 1 contient la valeur basse de l'adresse à laquelle on trouve le contenu de la variable. AD + 2 contient la valeur haute de cette même adresse.

Une variable tableau

La partie donnée d'un variable tableau est mémorisée de façon identique à celle d'une variable simple. Les éléments sont simplement disposés les uns après les autres en commençant par l'élément d'indice 0.

Si la variable a une seule dimension (vecteur) :
Si AD est l'adresse du pointeur de l'élément var (0). Alors: AD-1 et AD-2 contiennent le nombre d'éléments du vecteur. Autrement dit la dimension + 1. Ce nombre codé sur 16 bits est présenté avec l'octet le moins significatif en AD-2 et l'octet le plus significatif en AD-1. AD-3 contient la dimension de la véritable (1).

AD-4 et AD-5 contiennent le nombre d'octets à ajouter à AD-4 pour arriver à la variable suivante (OFFSET). Autrement dit AD-4 et AD-5 contiennent 3 + N × L avec L égal au nombre d'éléments et N égal à 2,3, ou 5 suivant le type de variable. AD-4 contient l'octet le plus significatif de ce nombre et AD-5 contient l'octet le moins significatif.

AD-6 contient le type de variable (1,2 ou 4).

AD-7 contient la valeur ASCII de la dernière lettre du nom de la variable augmentée de 80H.

AD-8 contient l'avant dernière lettre du nom de la variable et ainsi de suite. Exemple variable entière ALBA dimensionnée à 8: DIM ALBA (8)

AD-10	AD-9	AD-8	AD-7	AD-6	AD-5	AD-4	AD-3	AD-2	AD-1	AD	AD+1
A	L	B	.	.	.	.	.	.	.	.	.
41	4C	42	C1	1	13	0	1	9	0	valeur	0

Si la variable est multidimensionnée la structure est la suivante:
AD-n contient la valeur ASCII du dernier caractère du nom de la variable augmenté de 80H. AD-n + 1 contient le type de la variable.

AD-n + 2 et AD-n + 3 contiennent la longueur de l'OFFSET (1 + 2 × nombre d'indices + N × L).

AD-n + 4 contient le nombre d'indices.

AD-n + 5 et AD-n + 6 contiennent le nombre d'éléments du premier indice.

AD-n + 7 et AD-n + 8 contiennent le nombre d'éléments du deuxième indice et ainsi de suite.

Remarques: pour déterminer n il suffit de faire 6 + 2 * nombre d'indices. La description des variables tableau à une dimension est un simple cas particulier de la description des variables multidimensions.

Daniel Martin



Une visite chez Amstrad France



Marion Vannier et ses collaborateurs dans le feu de l'action

Amstrad France a récemment changé de locaux. Si un déménagement pose toujours les mêmes problêmes, dans ce cas précis ce n'était pas à cause de la distance. Amstrad s'est installé quelques centaines de mètres plus bas, dans la même rue.

Au cinquième étage on trouve les bureaux occupés par Amstrad. Lorsque l'on arrive, c'est tout d'abord Isabelle la jeune et jolie standardiste-hôtesse d'accueil qui vous reçoit. On devine pourquoi il est parfois difficile de joindre une personne, si les interlocuteurs connaissent leur correspondante.

La dame de fer

Comme les locaux sont grands on a une impression de calme, de mécanique trop huilée, d'administration feutrée. En fait il n'en est rien, c'est une rûche, où chacun à son poste permet à l'ensemble de fonctionner sans difficultés.

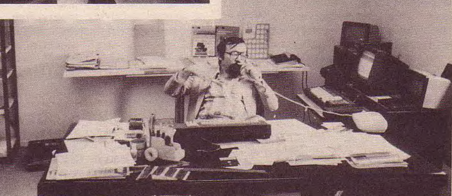
Ici, le vrai patron c'est Marion Vannier, directrice générale d'Amstrad France. Pendant la séance de photos dans son bureau, coups de téléphones et collaborateurs n'ont pas cessé de se succéder. Il faut avoir une santé de fer pour tenir un rythme semblable. Le succès c'est aussi de pouvoir ''assurer'', c'est beaucoup de travail.

Dans un bureau proche de celui du ''patron'' on peut rencontre Jean Cordier, chef des ventes. C'est grâce à lui si vous trouvez un CPC dans n'importe quelle ville de France. Sur les murs sont installées des cartes géographiques qui servent à repérer, grâce à des épingles de couleurs, la couverture commerciale d'Amstrad. Telle couleur est affectée aux représentants, telle autre aux grossistes. Actuellement J. Cordier installe une structure de service après-vente performante, qui se traduit déjà par près d'une vingtaine d'ateliers de réparations, ouvert aux professionnels, avec des techniciens de pointes et véritablement spécialistes des ''CPC''.

Dans une vaste salle sont disposés plusieurs bureaux autour desquels sont installés les ''administratifs'' de la société. Trois femmes et un homme pour s'occuper de tous les problèmes de comptabilité, des expéditions, de la douane, des litiges avec les particuliers. Eux aussi assurent dans une très large mesure le succès d'Amstrad.

L'Amstrad, ils connaissent

Un peu plus loin et légèrement isolé, le bureau du chef de produit, François Quentin. Si vous avez rencontré un problème technique avec votre CPC, vous avez certainement dialogué avec lui. Il connaît ''sa'' machine, n'essayez pas de lui faire la leçon, vous avez toutes les chances de vous casser les dents. Comme la plupart de membres d'Amstrad France, il passe son temps à régler les problèmes par téléphone, quelquefois avec des interlocuteurs à qui il faut expliquer en détail comment simplement charger une cassette avec un 464. Dur, dur !



Le chef de produit, François Quentin

Il est heureusement assisté par deux collaborateurs compétents et, eux aussi, d'une patience d'ange, Eric Zingraff et Hélène Dennery. C'est également cette équipe qui vérifie les manuels, les logiciels, traduit les documentations.

Nous sommes partis discrètement sur la pointe des pieds pour laisser tout ce petit monde travailler, et continuer d'assurer le succès de notre micro préféré.

Ph. Lamigeon



ERE informatique contamine le monde !

ERE informatique lance un nouveau jeu sur Amstrad : Contamination. C'est en exclusivité pour notre magazine, que l'équipe conceptrice a dévoilé les secrets de son nouveau logiciel.



A l'origine le projet était totalement différent, car il ne s'agissait pas de concevoir un nouveau jeu mais un logiciel, commandé par la chaîne de télévision TF 1. Cette dernière avait effectivement demandée à ERE de concevoir un logiciel graphique sur MO 5, représentant une mappemonde qui permettrait de suivre le vol de la navette spatiale américaine, à bord de laquelle devait être embarqué le spacionaute français Patrick Baudry. Or, pour des raisons techniques, le vol fut annulé et reporté. De ce fait, tout le monde oublia le logiciel, tant du côté du TF 1 que de ERE. Pourtant un jour, quelqu'un ressortit le projet. Que faire d'une belle carte en couleur ? Et pourquoi pas... On retrouve donc à la base du jeu toute l'équipe de ERE : Emmanuel Viau, Marc-André Rampon, Philippe Ulriche et Michel Rho.

Sauveur du monde

L'idée de départ, c'était bien entendu de conserver la carte. Il fallut donc travailler autour de cette structure de base et développer une histoire prenante. En entrant des données sur la situation géopolitique des pays, on pouvait grâce au curseur (en le positionnant quelque part sur la carte) obtenir le nom du pays, le nombre d'habitants etc. Le logiciel était intéressant, mais trop proche du style éducatif. Par contre, avec quelques notions de biologie molléculaire, vous pouvez fabriquer de jolies "bébêtes" qui contaminent en quelques minutes un pays, voir un continent. Et avec cet élément supplémentaire, vous obtenez le thème d'un jeu passionnant.

Vous pourriez facilement dominer le monde si vous aviez ce pouvoir de contamination. En fait, votre rôle est différent. Il vous faut sauver avec les moyens mis à votre disposition les pays touchés aléatoirement. Pour cela vous disposez d'une solution radicale : l'arme nucléaire. A n'utiliser qu'en cas de désespoir total. Vous pouvez également brûler une zone dangereuse ou encore dresser un cordon sanitaire, pratique assez efficace mais à très court terme. La meilleure solution est de prélever un échantillon, d'en effectuer l'analyse au laboratoire central et d'essayer de trouver l'antidote. Pour celà, vous pourrez visualiser le virus incriminé. Cela semble être la plus longue des possibilités mais c'est certainement la meilleure. Toutes les fonctions seront sélectionnables à partir d'icônes disposées au bas de l'écran.

Les réalisateurs du logiciel

Les dessins sont confiés à Michel Rho, ancien dessinateur-illustrateur pour la presse et la mode, graphiste chez ERE. C'est lui qui réalise les dessins de jaquettes et les pages de présentations des logiciels. La belle secrétaire d'Amélie Minuit ou le rocker de Macadam Bumper sont deux exemples de ses travaux. Son travail terminé, il transmet ses épreuves, obtenues avec l'aide d'un logiciel graphique, au programmeur qui devra les insérer dans son programme.

Le programmeur de ce jeu c'est Jean-Pierre Le·Clezio. Analyste-programmeur de métier, il passe ses loisirs à programmer. Mais oui ça existe ! La programmation est effectuée en assembleur, le jeu est conçu en temps réel. De toute façon, nous avons affaire à un spécialiste. Lui-même créateur de jeux, il vient de transposer "Mission Delta" sur Amstrad.

Pour finir, il fallait un peu de mùsique. On fit appel à Charles Callet, musicien de son état. Célèbre parce qu'il est l'auteur d'une chanson, interprétée par Eve Brenner, et dont le succès fut mondial. Il vient de terminer le dernier trente centimètres de Gérard Lenorman. Possésseur d'un Amstrad, il a créé un logiciel musical, qui sera édité par ERE. La musique qu'il a conçue pour "Contamination" est adaptée à la situation, tantôt gaie, tantôt franchement lugubre.

A la fois un fantastique jeu qui vous entraîne dans une aventure palpitante, et un bon moyen d'apprendre des tas de choses sur les pays, "Contamination" devrait au moins plaire à beaucoup sinon devenir un "best".

Philippe Lamigeon

VISMO Sélection

Vente Informations Services Micro-Ordinateurs

Micro-ordinateurs familiaux

84, bd Beaumarchais - 75011 Paris
Métro Bastille ou Chemin Vert
De 10 h à 20 h sauf dimanche.

338-60-00

22, bd de Reuilly - 75012 Paris
Métro Daumesnil ou Dugommier
De 14 h à 20 h sauf dimanche et lundi

VISM'EDIT

PRESENTE

COMPTAMSTRAD

LOGICIEL COMPTABILITÉ sur AMSTRAD :
Version cassette
200 Comptes 750 lignes d'écritures

Ce logiciel vous permet de gérer vos comptes suivant les indications ci-dessous. les 750 lignes d'écritures pouvant être définies par séquence soit mensuelle soit d'une durée réduite quinzaine, hebdomadaire, ou autre suivant le nombre d'écritures que vous avez à comptabiliser.
Vos comptes et leurs mouvements sont tenus en permanence dès que vous avez saisie toutes les écritures.

Vous pourrez donc à tout moment les consulter, tirer une balance, qui non seulement sera conforme au PLAN COMPTABLE mais qui de plus par une gestion de COLLECTIFS BILAN et GESTION sera un TABLEAU de BORD de votre ENTREPRISE à condition bien entendu que toutes vos écritures soient correctement passées ce qui ne dépendra que de vous.

Procédure de chargement

1. Presser simultanément la touche CTRL et la petite touche ENTER.
2. Presser la touche PLAY du lecteur de cassettes.
3. Appuyer sur une touche du clavier.



NOUS N'AVONS PAS TROUVE MIEUX

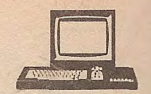
POURQUOI ATTENDRE ?

prix d'une vrai COMPTA

450,00 FRANCS TTC

COMMANDE A ADRESSER A VISMO INFORMATIQUE
VEUILLEZ LIBELLER VOTRE REGLEMENT A L'ORDRE DE VISMO

ENVOI EFFECTUER PAR CORRESPONDANCE EN 48 HEURES
SANS FRAIS DE PORT



LIBRE
EXPRESSION

Nous ouvrons la rubrique "libre expression" à la société Amstrad France. Celle-ci pourra intervenir régulièrement dans nos colonnes, pour s'exprimer sur un point précis concernant ses micros-ordinateurs, ou pour développer un thème libre. Pour inaugurer cette rubrique, Amsoft nous donne des précisions sur la compatibilité entre CPC 464 et CPC 664.

Le 664 incompatible !?

Nombreux d'entre vous ayant acquis un 664 se sont trouvés confrontés aux problèmes relatifs au chargement de certaines cassettes de jeux. On achète un jeu chez son revendeur, on jubile déjà rien qu'à l'idée de faire sauter le HI SCORE, on charge et c'est le plantage ou les messages d'erreur ou encore un logiciel inopérant : bref impossible de jouer avec les cassettes en question : toutes les ruses restent vaines. Où est la comptabilité ascendante dont nous parlait AMSTRAD ? Pourquoi ces logiciels ne passent pas ? Y-a-t-il une solution à ces problèmes ?

Les trois cas principalement constatés lors d'un problème de chargement d'une cassette sur CPC 664 sont les suivants:
- Après quelques minutes la machine tout à coup subit un reset général et vous envoie son message d'invite.
- Le programme se charge sans problème mais il est impossible d'aller plus loin que la première page du jeu qui vous convie de façon insistante à appuyer sur une touche pour commencer à jouer.
- Enfin dernier cas typique, un Memory full in XXX sournois vient interrompre le chargement du programme et vous ramène à la réalité d'une mémoire (semble-t-il) trop exigûe !

Les machines sont pourtant bel et bien compatibles et il ne faut s'en prendre qu'au soft. La cause des deux premiers cas d'incompatilité est la suivante : les développeurs de ces logiciels utilisent un adressage direct en ROM sans utiliser les routines copiées en RAM. L'adressage des routines en RAM est bien le même sur le 464 et sur le 664, ce qui les rend compatibles mais l'identicité de la ROM n'est pas une obligation de compatibilité (ce qui vous permet d'avoir un système dit d'architecture ouverte). Ainsi un adressage direct en ROM provoque parfois sur le 664 des effets particuliers, certaines maisons de soft n'ont pas respecté ou tenu compte de cette éventualité à l'époque où le 664 n'était pas encore annoncé. A 99,9 % les logiciels Amsoft n'occasionnent pas ces deux types de plantage, la raison étant bien évidemment le respect de la règle de l'adressage en RAM pour toute production. A l'heure actuelle rien n'est possible pour passer ses programmes sur CPC 664 sans modification du logiciel lui-même (ce qui ne peut être fait que par les concepteurs).

Mais reste le dernier cas : "Memory full in XXX". Vous vous en doutez tous il s'agit de la place mémoire prise par les routines d'exploitation du lecteur de disquettes qui abaisse le haut de la mémoire de 1284 octets. Autant il est simple sur un 464 de débrancher le contrôleur, autant la même opération est pour ainsi dire impossible sur un CPC 664. La routine ci-dessous est en fait l'interrupteur qui va vous permettre de transformer votre CPC 664 en un 464 en "déconnectant" le lecteur de disquette et de retrouver vos chers 43533 octets de mémoire. N'oubliez pas de sauver votre listing avant de l'exécuter : il s'agit de code machine ! Une fois le programme éxécuté, chargez sans problème tous les logiciels qui vous envoyaient le message de saturation.

```
190 MEMORY &7FFF
200 entry =&8000
210 sum =0
220 FOR addr=0 to &3E
230 READ byte$
240 POKE entry+addr, VAL("&"+byte$)
250 sum=sum+VAL("&"+byte$)
260 NEXT addr
270 IF sum<> 6058 THEN PRINT "Data error":END
280 CALL entry
290 NEW
300 PRINT HEX$(sum)
310 DATA cd,c8,bc,7a,b3,20,05,06,00,11,0
    6,c0,ed,53,3c,80
320 DATA ed, 43,3e,80,21,1c,80,0e,fc,cd,1
    6,bd,2a,3c,80
330 DATA ed,4b,3e,80,48,3e,c9,32,cb,bc,2
    2,39,80,79,32,3b,80
340 DATA 11,40,00,21,ff,ab,df,39,80,00,0
    0,00,00,00,00,00
```

Notez que certaines maisons de soft ont déjà une version pour les 664 des logiciels dont il est question. D'autre part cet état de fait ne devrait avoir qu'un temps puisque déjà les nouveaux logiciels sortis depuis deux mois sont parfaitement compatibles.

Amsoft

PETITES ANNONCES

Envoyez-nous vos P.A. **gratuites** rédigées **très lisiblement** à : Amstrad Magazine, 55, av. Jean Jaurès, 75019 Paris.

Suite achat groupé CPC 464/664, conditions très intéressantes (prix et garantie).
M. Bichaud, 20, rue Louis David, 93170 Bagnolet. Tél. 361.18.59, après 19 h.

Vds TI 99 4/A Péritel + joystick, câble magn., Basic, étend., Ext. 32 Ko, mini mém. + modules : foot, échecs, vidéogames 1, adventure (avec 3 K7) + K7 hebdogiciels n 1 à 4 + 15 progs, 7 revues et livres + 35 num. Hebdogiciels. Le tout 4.500 F.
Lamcelot, 3, rue des Juifs, 18000 Bourges. Tél. (48) 24.03.50 après 20 h.

Echange programmes CPC 464.
Frédéric Belfis, 10, av. Courbe, 06600 Antibes. Tél. (93) 33.60.16.

Vds progs initiation au Basic Amstrad, Easi-Amsalc, budget familial (Sprites), Amcalc (SDI) Ambase (SDI). Progs éduc. : lettres magiques, ardoise magique, l'Horloger un, Electro Freddy. Progs jeux : Harrier attack, Sultan's maze, Oh mumm.
Rech. tout logiciel de gestion préf. sur disq. et contact avec possesseur de lect. disq.
Dominique Blandin, 6, place du Pré Botin, 4 636, Chambpanoist, 77160 Provins. Tél. 067.60.19.

Echange progs sur 664 (liste importante) de préf. en lang. mach. Recherche astuces.
T. Sitruk, 37, rue Faidherbe, 75011 Paris. Tél. 371.58.15.

Vds TO7, 3 cartouches (Basic, Pictor, Trap), interf. jeu/son + manettes + surprises. Jamais servi. 2.000 F.
Mini chaîne hi-fi (tuner, platine disq., K7, ampli). T.B.E. Val. 1.600 F, vendu 800 F.
Tél. (68) 42.06.10.

Cherche pour 464/664 progs et interfaces pour décodage Morse-RTTY. Echange également jeux et utilitaires.
J. Parsi, 28, rue Taine, 75012 Paris. Tél. 341.03.37 après 20 h.

Vds Amstrad CPC 464, moniteur couleur, sous garantie + logiciels, jeux, livres. 3.800 F.
Tél. 851.92.86 après 17 h 30.

Vds Yeno SC 3000 (déc. 84) avec Péritel + K7 jeux + 120 progs + livres div. état neuf. Val. 4.000 F cédé 2.000 F.
J. Macri, 2, bd J. Rostand, 13920 St-Mitre les Remparts. Tél. (42) 80.95.89, le soir.

Introduction à la PROGRAMMATION

POUR BIEN DEBUTER

La programmation nécessite l'approche logique d'un problème de façon à le rendre, grâce à un langage reconnu par l'ordinateur, éxécutable. Cette logique de programmation est très éloignée du comportement naturel humain, qui ne se pose que trop rarement des questions sur ses agissements. Trop peu de programmeurs utilisent une méthode pour travailler. Il faut avouer que la plupart des livres traitant du sujet sont d'un abord difficile pour qui n'est pas mathématicien. L'initiation que vous proposons ici ne vous transformera pas en programmeur professionnel, mais vous permettra certainement de réaliser des programmes avec une meilleure démarche d'esprit, et vous donnera peut-être envie d'approfondir le sujet avec des ouvrages spécialisés.

La programmation, c'est facile!

Bien des possesseurs d'ordinateurs ont lu ou entendu ce slogan, qui fut parfois un élément déterminant dans leur décision d'acquérir une machine. Après avoir assimilé les notions de base, ils déchantèrent rapidement devant la complexité du problème. Ils se firent une raison bien amère, si l'on pense que certains d'entre eux ont dépensé des fortunes en matériels : programmer c'est très difficile !

Réaliser un programme de dix lignes, avec trois instructions de Basic, est à la portée d'un enfant de dix ans. Oui, mais le même programme peut s'éxécuter, parfois avec deux instructions en quatre lignes. Certains programmeurs amateurs s'attaquent dès leurs débuts à des problèmes très complexes, tel que des programmes de comptabilités, de jeux, aux algorithmes compliqués, qui ne tourneront avec un peu de chance, qu'après de nombreux mois d'efforts. Il est vrai qu'il faut un minimun de connaissances pour programmer. Mais plus encore que la connaissance profonde d'un langage, il faut possèder avant tout une "logique d'esprit", c'est-à-dire une méthodologie, un savoir faire. Il n'est pas nécessaire d'être logique pour se mettre à programmer, mais pour réussir un programme il faut le devenir.

Pourquoi faut-il programmer un ordinateur?

La réponse est simple : parce qu'il s'agit d'une machine et qu'il est donc obligatoire de guider son travail. Essayez même s'il s'agit du modèle le plus sophistiqué, de faire travailler votre aspirateur en lui indiquant simplement, et oralement les pièces de votre appartement, qu'il doit nettoyer. Quel résultat obtiendrez-vous ? Aucun ! Tout simplement parce que vous ne pouvez pas demander à une machine d'exécuter un travail qu'elle ne peut pas exécuter seule. D'après d'éminents "spécialistes", l'ordinateur serait une machine intelligente. Il s'avèrerait comme particulièrement intéressant qu'ils en fassent la démonstration. Car, même si un robot peut aujourd'hui jouer une partition musicale, il fut à l'origine programmé par un homme.

Il est possible de stocker sur une surface, qui se réduit de jour en jour, le contenu d'une encyclopédie. Mais qui affirmera connaître le fonctionnement du cerveau humain et le volume d'information conservé par celui-ci, éléments essentiels pour l'élaboration de machines dites intelligentes ? Ces dernières existent, mais leurs possibilités sont très réduites. Les tâches qu'elles sont capable d'exécuter se limitent à des travaux industriels, à la restitution d'informations traitées ou non, à l'assistance ou à l'enseignement de techniques de pointes. Elles ne sont en aucun cas capables de raisonnement naturel.

Il faut donc indiquer dans quel ordre les tâches doivent être exécutées, et quelles données utiliser. Ignorer l'intelligence artificielle semble être un comportement illogique pour un informaticien. Mais sachez que deux machines ne se feront jamais la guerre, sauf si l'homme les

programme à cet effet.

Connaître sa machine

Avant même d'entreprendre la réalisation d'un projet de programme, il faut connaître réellement les possibilités de son micro-ordinateur. Il sera particulièrement difficile de réaliser un logiciel qui fait appel constamment à des fichiers extérieurs, si l'ordinateur n'est pas équipé d'une unité de disquettes.

Par exemple, supposons que dans un programme on fasse appel à des informations se trouvant stockées dans un fichier clients, et qu'elles doivent être utilisées avec celles du fichier commandes. Les nombreuses manipulations qui se dérouleront pendant le traitement rendront concrètement le logiciel inutilisable. Il faudra charger la cassette fichier clients, lire le premier nom, charger la cassette fichier commandes, repérer le compte correspondant au client, éxécuter le traitement, sauvegarder enfin sur une cassette l'information ainsi traitée. Il n'est bien sûr aucunement interdit de travailler dans ces conditions, mais il sera probablement plus rapide de traiter ce type d'application avec des procédés classiques et manuels.

Par contre, l'unité de disquettes se justifie particulièrement dans des cas comme celui-ci. L'accès quasi-instantané qu'offre la disquette, permettra de travailler constamment sur des fichiers en entrée-sortie. Il ne faut pas non plus confondre les capacités de son micro-ordinateur avec celles qu'offrent les systèmes de catégorie professionnelle. Il est, comme vous pouvez le constater, particulièrement important de bien connaître sa machine avant d'entreprendre la réalisation d'applications qui, si elles apparaissent trop complexes par rapport à la configuration disponible, ne tourneront jamais ou alors avec d'énormes complications.

Savoir ce que l'on veut faire

Avant de commencer à programmer, il est souhaitable de savoir ce que l'on veut réaliser comme programme. Partir sur une idée et commencer sans plan de travail à entrer des instructions ne sert à rien. On peut être certain de s'engager sur la voie de l'échec avec ce type de démarche. L'utilisateur d'un micro-ordinateur n'a souvent de l'informatique qu'une vague idée. Il ne souhaite pas apprendre, il ne veut qu'utiliser et cela le plus rapidement possible.

Pourtant, avant d'acheter une nouvelle voiture on regarde les nouveaux modèles proposés par les constructeurs, on calcule la valeur du crédit, le prix des options et des accessoires. On établit une liste des voitures qui semblent correspondre à nos besoins et à nos moyens, puis on rend visite aux distributeurs. Si l'on est satisfait d'un modèle et que l'achat est effectué, c'est parce que dès le départ on connaissait son sujet. La réalisation d'un programme demande d'effectuer la même démarche d'esprit, c'est-à-dire de bien connaître le sujet à traiter. Une personne qui maitrise parfaitement un domaine, ne recontrera généralement pas de grandes difficultés.

Mais par contre, si l'on envisage de réaliser un simulateur de vol d'avion sans connaître simplement les instruments et leurs utilisations, il est fort probable d'aboutir sur un échec. Malgré tout, en utilisant à bon escient toute la documentation concernant ce sujet, et avec l'aide d'un pilote d'avion, il est possible d'obtenir des résultats très spectaculaires. On peut normalement tout programmer, en connaissant au moins ses limites sur le problème posé. La documentation est une chose primordiale, car elle permettra de contrôler ses connaissances et d'établir un plan de travail détaillé, avec pour principale qualité de vérifier que l'entreprise n'est pas trop grosse. On peut nommer cette partie: Analyse de base.

La programmation

La programmation consiste à décrire très précisément chacune des tâches que l'ordinateur devra éxécuter pour fournir un résultat, ceci avec l'aide d'un langage de programmation. Par exemple, on doit indiquer clairement la forme des enregistrements d'un fichier : Nom, Prénom, Numéro de Téléphone, Adresse. De plus, le programme, devra connaître la provenance des informations : est-ce l'utilisateur qui les fournira à partir du clavier, devra-t-il aller les chercher sur un fichier cassette ? Leurs destinations également : où ranger l'information traitée ? Si une tâche est oubliée, ou mal expliquée, le programme se "plantera". Toutes les phases du traitement devront être abordées avec le même raisonnement : détailler à l'extrême chaque tâche, de façon à assurer au logiciel des chances de "tourner" rapidement, et d'éviter ainsi à peiner longtemps pour des résultats médiocres, pour être dégouté rapidement de la programmation.

Structurer le programme

Pour qu'un programme soit à la fois court et performant, il vaut mieux l'optimiser au maximun. Pour cela il faut lui donner une structure logique. Plus explicitement, si dans un programme nous devons transférer certaines informations afin qu'elles puissent être traitées ensemble, il est souhaitable de regrouper les opérations de transfert, dans la mesure du possible, afin que ce transfert s'effectue en une seule opération.

Reprenons l'exemple précédent: nous devons traiter les données contenues dans le fichier "Adresses", il est préférable de charger en une unique opération le nom, le prénom, le numéro de téléphone, l'adresse que d'avoir à appeler chaque information séparement. Il sera également plus facile de repérer ses erreurs de programmation et donc de dépanner un programme structuré, qu'une suite illogique et anarchique d'instructions. Enfin le programme sera plus court, donc un gain important de place en mémoire.

La constitution du dossier

Après avoir réuni une documentation sur le sujet, que l'on souhaite programmer, il est particulièrement intéressant de se constituer un dossier de programmation. Nous allons donc examiner le contenu de ce dossier, qui sera très utile si l'on veut actualiser son programme, le modifier ou encore l'adapter pour une autre machine.

L'organigramme

Souvent négligé, voir inconnu de beaucoup de programmeurs, l'organigramme est pourtant un document bien utile, qui peut rendre de nombreux services. Il facilite notamment les vérifica-

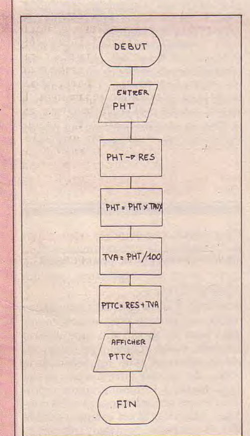
tions pour contrôler le fonctionnement d'un projet de programme. A la base de l'organigramme, il y a un algorithme. Une série d'opérations ou de calculs réalisés selon un procédé précis et uniforme afin d'obtenir la solution d'un problème donné, est un algorithme.

Exemple: un commerçant doit calculer le prix TTC d'un produit. L'algorithme, qui permet d'obtenir ce prix se décompose en une suite d'opérations qui donneront le résultat attendu:

$$PTTC = \frac{PHT * taux\ TVA}{100}$$

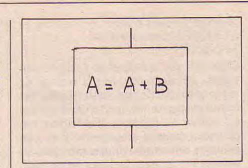
1. donner le prix hors taxe du produit.
2. multiplier le prix hors taxe par le taux de TVA
3. diviser par 100 le résultat, donnant la valeur de la TVA.
4. additionner le PHT avec la TVA.
5. Afficher le prix TTC.

L'organigramme qui correspondant à cet exemple sera:

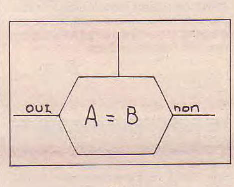


L'organigramme est donc la représentation shématique d'un algorithme. Un organigramme utilise des figures géométriques pour indiquer le travail à réaliser.

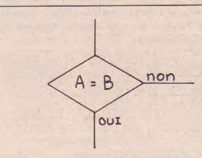
Le rectangle contient une opération de traitement, ou le report à un traitement.



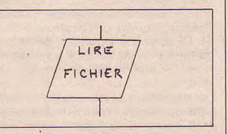
• Les tests sont représentés par:



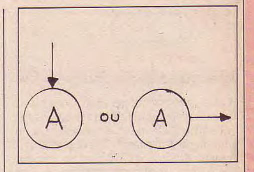
OU



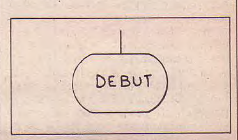
Lorsque l'organigramme est trop chargé, on utilise les cercles comme connecteurs, pour le renvoi à certains points du shéma.



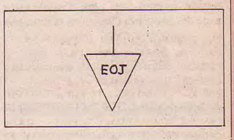
Le parallélogramme indique les entrées/sorties, lecture et écriture.



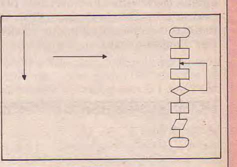
Le début et la fin du traitement sont précisés par:



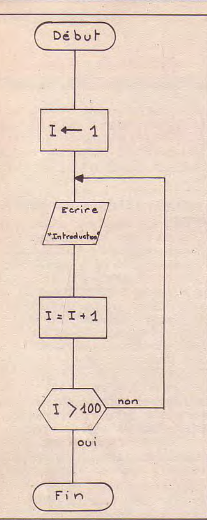
Le triangle est utilisé pour une intérruption dans le traitement, et spécifié par E.O.J. (End Of Job).



Les segments de droites avec, ou sans, flèches représentent le cheminement de l'organigramme.



Voici un autre exemple d'utilisation d'un organigramme: le programme shématise ici, affiche 100 fois sur l'écran le mot "Introduction".



Inventer son langage

Comme vous pouvez facilement le constater, l'organigramme permet de représenter shématiquement le cheminement du programme, mais il ne détaille pas encore suffisamment les opérations. Il est pratique de le compléter par une description méticuleuse du programme, en s'aidant d'un pseudo-langage. Dans la pratique, il est possible d'utiliser soit les deux "outils", soit l'un d'entre eux.

Reprenons l'exemple précédent. Il fallait afficher 100 fois le mot "Introduction" sur l'écran, grâce à l'utilisation d'une boucle. Ecrit en pseudo-langage le détail donnerait:
Mettre 1 dans I
Ecrire "Introduction"
Si I est plus petit que 100 alors I = I + 1, sinon arrêter le traitement.
Aller à 10

Il est donc ensuite très aisé d'écrire le petit programme qui correspond à cette description.

```
10 FOR I=1 TO 100
20 PRINT "Introduction"
30 NEXT I
```

Créer un pseudo-langage (ou pseudo-code), ne demande que l'emploi de mots-codes facilement assimilables par toutes les personnes qui seront amenées à l'utiliser. L'échange de programmes écrits en pseudo-langage ne pose, en principe aucun problème. De plus, ils offrent l'avantage de décrire un travail à éxécuter, mais sans tenir compte du langage employé. Concrètement, un programmeur qui utilise le Basic peut transmettre à un ami son dossier qui lui, programmera en Pascal. Outre les facilités d'échanges, la maintenance du programme est abordable après plusieurs mois de fonctionnement

Les mots-clés d'un pseudo-langage

Il est plus simple d'utiliser la langue française pour "fabriquer" son langage. Nous vous présenterons quelques mots, mais libre à vous de le réaliser en Anglais, en signes mathématiques ou en Serbo-Croate. Attention toutefois si vous envisagez de travailler en groupe, il faudra ensuite que les autres vous comprennent.

Le verbe METTRE ou la flèche → affecteront une valeur dans un registre ou une zone réservée.
METTRE 1 dans A (A←1).
Avec AFFICHER OU ECRIRE, on obtiendra soit la visualisation d'un résultat soit son impression :
ECRIRE solde compte

Les conditions sont représentables par SI.. ALORS.. SINON.
SI deux rouge ALORS gagné SINON perdu
Pour indiquer un branchement on peut utiliser ALLER
ALLER calcule des racines
La répétition se formulera TANT QUE ou REVENIR.
TANT QUE boule blanche ALLER choix
REVENIR choix si boule blanche
On conservera le OU et le ET logique.
Si a OU b sont égal à 100 ALLER lecture

Les instructions de sauvegarde, de chargement, d'ouverture et de fermeture de programmes et de fichiers seront très simples, comme : SAUVE, CHARGE, OUVRE, FERME.
SAUVE programme
OUVRE compte

La lecture est exprimée par LIT
LIT fichier adresses.

Un debogger manuel

On pourrait donner en détail, et mieux que cela, une suite de mots codes. Mais comme nous l'avons déjà dit précédemment, chacun invente son propre langage comme il le souhaite. Avec ces quelques mots et un organigramme, on peut écrire des programmes très compliqués avec une possibilité de contrôle des erreurs, de logiques ou d'écritures très importantes. En effet, l'utilisation de ces outils d'aide à la programmation, permettent de "faire tourner" le programme à la main.

Cette vérification est très simple, puisqu'il suffit d'une feuille de papier et d'un crayon. On dessine très fidèlement la forme des enregistrements utilisés, le nom des registres, puis on "descend" le programme en notant très scrupuleusement tous les transferts, les résultats des calculs et ce qui s'imprime ou s'affiche à l'écran. Ce travail est long, mais garantit en principe le bon fonctionnement du programme, grâce à la découverte des erreurs qui ont pu être "programmées" en pseudo-langage.

Un exemple simple

Pour mieux comprendre, nous allons voir ensemble la programmation d'un algorithme simple en pseudo-langage, après avoir été représenté par un organigramme.

Beaucoup d'entre nous utilisent chaque matin une voiture pour se rendre sur leurs lieux de travail. En quittant son domicile, est-on vraiment sûr de retrouver sa voiture, d'avoir ses clefs dans la poche, de pouvoir ouvrir la portière et de démarrer ? L'organigramme représente toutes ces opérations avec les contrôles adéquats. Si cet exemple semble bête, il symbolise pourtant, très superficiellement, il est vrai, un travail que nous effectuons chaque jour sans réfléchir.

Pourtant nous agissons logiquement en effectuant les opérations dans un ordre parfait. Impossible de s'installer dans la voiture si la porte n'a pas été ouverte. Avec l'aide de l'organigramme (voir

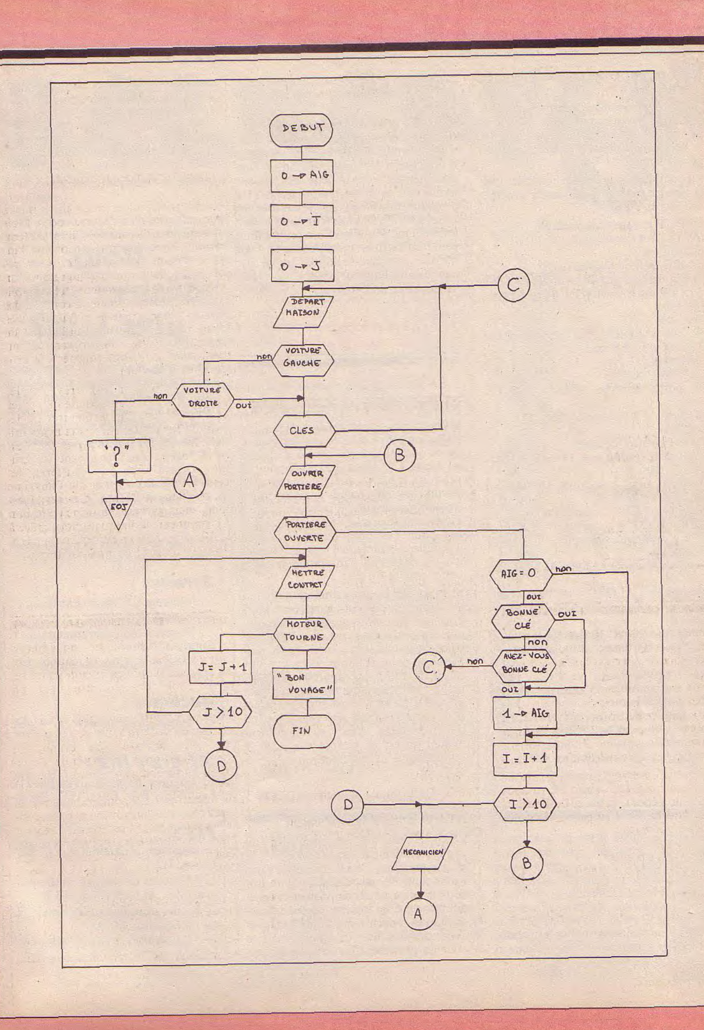

DEBUT
0 → AIG
0 → I
0 → J
DEPART MAISON
VOITURE GAUCHE
non
VOITURE DROITE
non
oui
CLES
"?"
EOJ
A
B
C
OUVRIR PORTIERE
PORTIERE OUVERTE
METTRE CONTACT
MOTEUR TOURNE
J = J + 1
J > 10
D
"BON VOYAGE"
FIN
AIG = 0
non
oui
BONNE CLÉ
oui
non
AVEZ-VOUS BONNE CLÉ
non
oui
1 → AIG
I = I + 1
I > 10
MECANICIEN

fig. 3) on peut écrire en pseudo-langage le programme.

"aig" *est un aiguillage pour le saut* d'un test après premier passage.
"i" et "j" sont des compteurs évitant le bouclage lors de la rencontre d'un problème.

En quittant la maison, on considère que la voiture se trouve soit sur la gauche, soit sur la droite.

DEBUT programme exemple
METTRE O dans aig,i,j
maison
FAIRE *quitter maison*
SI voiture à gauche OU à droite ALORS ALLER test clés SINON voiture disparue EOJ
Test clés
Si clés ALORS ALLER ouverture SINON ALLER maison
ouverture
OUVRIR porte
SI porte ouverte ALORS ALLER contact.
SINON ALLER bonne clé
contact
FAIRE mettre contact
SI moteur tourne pas ALORS AJOUTER 1 dans j
SI j pas plus grand que 10 ALORS ALLER contact.
SINON ALLER mécanicien
AFFICHER "bon voyage"
FIN programme exemple
Bonne clé
SI aig différent O ALORS ALLER plus
SINON
SI bonne clé ALORS METTRE 1 dans aig
SINON
SI clé absente ALORS ALLER maison
SINON METTRE 1 dans aig
plus
AJOUTER 1 dans i
SI i pas plus grand que 10 ALORS ALLER ouverture
(on peut aussi utiliser : TANT QUE i pas plus que 10 ALLER ouverture)
mécanicien
FAIRE appeler mécanicien EOJ.

Le traitement des erreurs

Durant la saisie d'un programme il arrive de frapper des mauvais caractères, qui vont générer des erreurs lorsque l'on lancera le programme. Ces fautes de frappes sont facilement décelables et *donc corrigeables. Après cette première* correction, il arrive que des erreurs dans l'éxécution du programme se produisent. Elles peuvent avoir pour origine un algorithme déffectueux, ou une erreur dans la conception du programme dès l'analyse.

Pour ces deux cas la programmation structurée facilite grandement la recherche des erreurs. De plus si l'on dispose d'un dossier de programmation bien documenté, le problême est encore moins délicat. Il suffit de reprendre les algorithmes utilisés, ou de vérifier les zones que l'on a réservées pour traiter les données. Par contre, un programme peut parfaitement tourner avec un "jeu d'essai" et se planter un jour sur un cas, que le programmeur n'aura pas prévu au moment de l'analyse.

La maintenance des programmes

Il est quelquefois nécessaire d'actualiser ou de modifier un programme pour diverses raisons. C'est souvent le cas des programmes de gestion dont la maintenance doit être assurée régulièrement, parce que tel taux de remise a changé ou que la TVA passe à 33,33% pour tel produit. Il arrive que par manque de documentation, et après plusieurs mois, on soit incapable d'intervenir pour modifier le logiciel. C'est notamment le cas pour des programmes écrits en assembleur, dans lequel le programmeur n'a pas jugé bon d'insérer le moindre commentaire.

Si un dossier de programmation n'a pas été constitué, on peut commencer à réécrire le programme en entier. La moindre des précautions est de garder une trace de ce que l'on a réalisé pour soi-même ou pour les autres. D'où l'importance de faire une analyse détaillée, inclure un organigramme ou une programmation en pseudo-langage (ou les deux) et le listing de la dernière version du programme. Avec ces deux documents, on est en mesure d'assurer la maintenance de ses créations.

N'oubliez pas l'utilisateur

Enfin, il reste une personne que l'on néglige totalement dans le mode de la programmation, c'est l'utilisation. Consommateur de nos réalisations, il doit se débrouiller pour comprendre le fonc*tionnement* d'un *logiciel, faute de docu*mentation. Fournir une feuille de papier avec quelques mots incompréhensibles pour un profane, c'est condamner son travail au plus doux des anonymats. D'aucuns se plaignent de cette situation surtout dans les milieux professionnels. Est-il profitable de posséder le meilleur logiciel de comptabilité, accompagné d'une documentation fantôme, que d'un produit moyen mais avec un manuel d'utilisateur digne de ce nom ?

Michel Merlet
Eric Charton

Vous désirez aller plus loin?

Pour cela, il existe un grand nombre de livres traitant de la logique de programmation. Nous vous proposons une liste, très sommaire, d'ouvrages spécialisés qui vous permettront d'approfondir le sujet.

Cedic :

☐ **L'algorithmique, de la pratique à la théorie,** J. Vicard, G. Chaty. 150 F.

☐ **Premières leçons de programmation.** J. Arsac. 100 F.

Eyrolles :

☐ **Initiation à la programmation.** C. Delannoy. 90 F.

☐ **Eléments de programmation.** J. Smith. 155 F.

☐ **Programmation avancée.** J.C. Broussard, R. Mahl. 120 F.

Hachette :

☐ **Savoir programmer.** J. Inglis. 120 F.

Mc Graw Hill :

☐ **Logique de programmation.** J. Tramblay, R.B. Bunt, P.G. Sorenson.

PSI :

☐ **Comment programmer.** J.C. Barbance. 105 F.

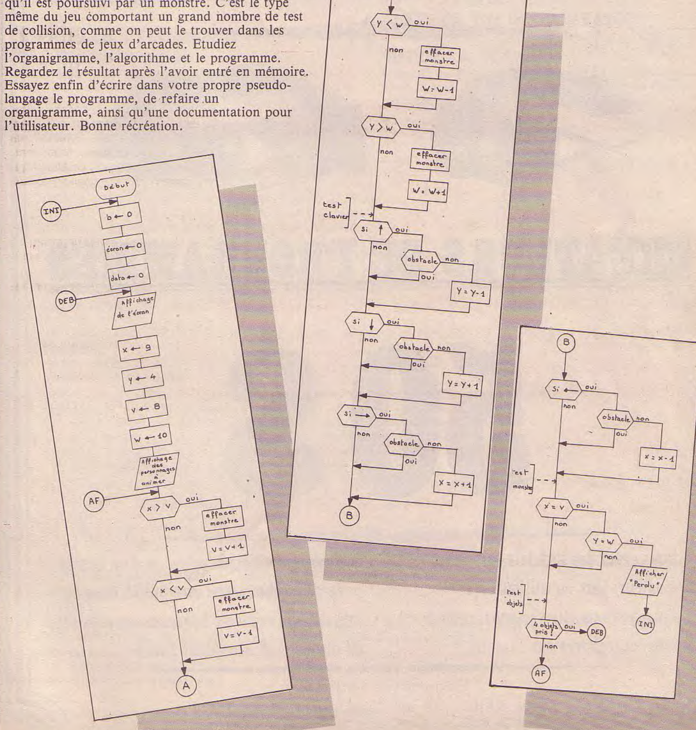
☐ Méthodes de réalisation des programmes. M. Benelfoul. 85 F.

☐ **La programmation inventive.** X. de la Tullaye. 100 F.

☐ **Technique de programmation des jeux.** X. Levy. 120 F.

RECREATION

L'organigramme et le programme suivants vous permettront de réaliser un petit jeu. Vous devrez aider le personnage à ramasser quatre objets, alors qu'il est poursuivi par un monstre. C'est le type même du jeu comportant un grand nombre de test de collision, comme on peut le trouver dans les programmes de jeux d'arcades. Etudiez l'organigramme, l'algorithme et le programme. Regardez le résultat après l'avoir entré en mémoire. Essayez enfin d'écrire dans votre propre pseudo-langage le programme, de refaire un organigramme, ainsi qu'une documentation pour l'utilisateur. Bonne récréation.



EXPLICATIONS:

DEBUT :

• Initialisation des variables de l'écran et du tableau.

AFFICHAGES :

• Affichage de l'écran et initialisation des variables de position des différents objets (x et y pour le petit personnage ainsi que v et w pour le monstre).

AFFICHAGE ET ANIMATION DES PERSONNAGES :

• En premier lieu test de position afin de réaliser une poursuite. Les variables du monstre sont incrémentées ou décrémentées en fonction de la position du petit bonhomme.

TEST CLAVIER :

• Le test est de la forme : si action alors vérifier s'il n'y a pas d'obstacle à la position d'arrivée oui, alors saut vers b non, actualiser x et y.

TESTS DE COLLISION :

• Ce test est de la forme si x = y , alors véifier que y = w et si y = w alors fin du jeu et retour à d.

TESTS DE FIN DE TABLEAUX

• Si quatre objets sont pris alors le tableau est fini, retour à c.

FIN DE LA BOUCLE :

• Retour au début, c'est-à-dire vers l'affichage des personnages à animer (l'affichage est réactualisé en fonction des variables).

```
1 REM listing amstrad
10 DATA 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
20 DATA 1,2,0,0,0,0,0,0,0,0,0,0,1,1,2,1
30 DATA 1,0,1,1,1,1,0,1,1,1,1,1,0,0,0,0,1
40 DATA 1,0,1,0,0,0,0,0,0,0,0,1,1,1,1,0,1
50 DATA 1,0,0,0,1,0,1,1,1,1,0,0,0,0,0,0,1
60 DATA 1,0,1,0,1,0,1,0,0,0,0,1,1,0,1,0,1
70 DATA 1,1,1,0,1,0,0,0,1,0,1,1,1,0,1,0,1
80 DATA 1,0,0,0,0,0,1,0,1,0,0,0,0,0,1,0,1
90 DATA 1,0,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1
100 DATA 1,2,0,0,0,0,0,4,0,0,0,0,0,0,0,2,1
110 DATA 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
120 CLEAR:MODE 0: RESTORE:DIM b(17,11):FOR y =1 TO 11:FOR x=1 TO 17:LOCATE x,y:R
EAD a:b(x,y)=a
130 IF a=1 THEN PRINT CHR$(206):ELSE IF a=0 THEN PRINT " ":ELSE IF a=2 THEN PRIN
T CHR$(230)
140 NEXT x:NEXT y
150 x=9:y=4:v=8:w=10
160 LOCATE x,y:PRINT CHR$(249):LOCATE V,W:PRINT CHR$(225):B(V,W)=4
170 U=U+1:IF U=2 THEN U=0:ELSE GOTO 220
180 IF X>V THEN IF B(V+1,W)=0 THEN GOSUB 290:V=V+1
190 IF X<V THEN IF B(V-1,W)=0 THEN GOSUB 290:V=V-1
200 IF Y<W THEN IF B(V,W-1)=0 THEN GOSUB 290:W=W-1
210 IF Y>W THEN IF B(V,W+1)=0 THEN GOSUB 290:W=W+1
220 IF INKEY(2)=0 THEN IF b(x,y+1)=0 THEN LOCATE x,y:PRINT " ": y=y+1:ELSE IF b(
x,y+1)=4 THEN 290:ELSE IF b(x,y+1)=2 THEN b(x,y+1)=0
:A=50:C=C+1
230 IF INKEY(0)=0 THEN IF b(x,y-1)=0 THEN LOCATE x,y:PRINT " ":y=y-1 :ELSE IF b(
x,y+1)=4 THEN 300:ELSE IF b(x,y-1)=2 THEN b(x,y-1)=0
:A=50:C=C+1
240 IF INKEY(8)=0 THEN IF b(x-1,y)=0 THEN LOCATE x,y:PRINT " ":x=x-1:ELSE IF b(x
-1,y)=4 THEN 300:ELSE IF b(x-1,y)=2 THEN b(x-1,y)=0:
A=50:C=C+1
250 IF INKEY(1)=0 THEN IF b(x+1,y)=0 THEN LOCATE x,y:PRINT " ":x=x+1:ELSE IF b(x
+1,y)=4 THEN 300:ELSE IF b(x+1,y)=2 THEN B(x+1,y)=0:
A=50:C=C+1
260 IF C=4 THEN RUN
270 IF B(X,Y)=4 THEN 300
280 GOTO 160
290 LOCATE V,W:PRINT " ":B(V,W)=0:RETURN
300 FOR X=100 TO 200:SOUND 1,X,2:SOUND 2,X,2:NEXT:LOCATE 5,25:PRINT"PERDU":FOR X
=1 TO 2000:NEXT :RUN
```

Garfunkel

Le premier programme est simplement un programme de présentation pendant le chargement du programme principal. Attention si vous estimiez inutile cette première partie, c'est raté ! Elle comprend certaines variables et commandes utilisées dans la seconde partie ; c'est pourquoi, à la ligne 770 vous avez la commande CHAIN MERGE qui permet de conserver les deux programmes en même temps. Avec cette commande, "DELETE 770" permet, en cas de "Break" de redémarrer sans recharger une nouvelle fois le programme principal.

Quand le programme principal commence, il vous donne d'abord le menu. L'utilisation d'INKEY plutôt que d'INPUT permet d'éviter de taper ENTER après le choix. Une colorisation facile du menu est obtenue par la mise en place de 6 fenêtres (ligne 1220).

Le choix 1 vous emmène à un sous-programme qui vous donne la règle du jeu, puis retour au menu après une temporisation ou en tapant "ENTER".

Les trois choix suivants vous permettent de choisir les touches d'utilisation ou le joystick. Un astérisque clignotant vous indique le choix sélectionné.

Les deux dernières options sélectionnent le jeu contre l'ordinateur ou celui à deux joueurs.

L'affichage du jeu, qui commence ligne 2290 après la redéfinition des caractères, se fait obligatoirement en mode 0 puisqu'un minimun de cinq couleurs est nécessaire.

Le jeu proprement dit débute ligne 2680. Les deux versions sont distinguées par la variable jb (jb = 1 pour un joueur, jb = 2 pour deux joueurs). La variable jr = 1 indique le premier joueur ou l'ordinateur. tt donne le temps de réponse calculé selon le nombre de notes hw. ha (hj) donne la valeur de la note qui est choisie aléatoirement entre 1 et 4 par l'ordinateur pour le jeu seul. Selon cette note un branchement est effectué ligne 2970 vers les sous-programmes son et affichage du logo. Si la note est fausse vous êtes dirigé vers le sous-programme "PERDU" avec affichage des scores.

Roger Rosec

LISTE DES VARIABLES

Par ordre d'entrée dans le listing

ag1$	: Caracteres ligne du haut du logo mode 1
ag2$	: Caracteres ligne du milieu du logo mode 1
ag3$	: Caracteres ligne du bas du logo mode 1
at$	: Nom de votre magazine prefere
ap1 a ap5	: Tracage de at$
aa$(1 a 5)	: Pour dessin du titre Garfunkel
ta$(1 a 5)	: Les 5 lignes du trace de Garfunkel
j1$ a j5$	: Les touches utilisees pour jouer
man	: Pointeur du menu 2, 3 ou 4. Emplacement asterisque
m	: Les notes de la musique d'accompagnement
mu	: Compte le nombre de notes jouees
jb	: =1 , 1 joueur - =2 , 2 joueurs
jj$(1)	: Nom du premier joueur
jj$(2)	: Nom du second joueur
tr	: Pointeur joueur qui commence
dar	: +240 = caractere dessine
dt, dg	: Position de ces caracteres
dar2	: Autres caracteres dessines
dv, dg2	: Position de ces caracteres
za, zb	: Position du logo
pp1$	: 1ere ligne du mot RECORD
pp2$	: 2eme ligne du mot RECORD
pp3$	: 1ere ligne du mot SCORE
pp4$	: 2eme ligne du mot SCORE
ttt	: Temporisation debut musique d'accompagnement
hw	: Nombre de notes a jouer
hj	: Numero de la note jouee
jr	: Joueur. Si 1 : premier joueur ou ordinateur
tt	: Nombre de secondes restant a jouer
ha(hj)	: Valeur de la note hj (de 1 a 4)
tp	: Longueur de la note jouee
note	: Valeur de note jouee
sc	: Score
st	: Record
sc1	: Record du joueur 1
sc2	: Record du joueur 2
sc11,sc12	: Position d'edition des scores
r$(1 a 16)	: Texte des regles du jeu

```
100 REM ********************************
110 REM ****  GARFUNKEL  ***** Vers. 1,2
120 REM ********************************
130 REM *** par Roger Rosec * Juillet 85
140 REM ********************************
150 REM >>>>>>> PREMIER PROGRAMME <<<<<<
160 REM ________________________________
170 MODE 1:CLS:DEFINT A-Z
180 ENT 1,100,2,2 :ENV 1,10,5,3
190 ENV 2,127,30,1,127,30,1
200 INK 0,16:INK 1,1:INK 2,3:INK 3,26,6
210 BORDER 16 :PEN 2:PAPER 0
220 SYMBOL AFTER 199
230 SYMBOL 230,255,128,128,128,143,143,143,142
240 SYMBOL 231,255,0,0,0,249,249,249,57
250 SYMBOL 232,255,1,1,1,249,249,249,193
260 SYMBOL 233,142,142,142,142,142,142,142,142
270 SYMBOL 234,1,1,1,127,127,127,57,57
280 SYMBOL 235,193,193,193,249,249,249,193,193
290 SYMBOL 236,142,143,143,143,128,128,128,255
300 SYMBOL 237,57,249,249,249,0,0,0,255
310 SYMBOL 238,193,193,193,193,1,1,1,255
320 ag1$=CHR$(230)+CHR$(231)+CHR$(232)
330 ag2$=CHR$(233)+CHR$(234)+CHR$(235)
340 ag3$=CHR$(236)+CHR$(237)+CHR$(238)
350 FOR i=21 TO 1 STEP -4:FOR j=2 TO 38 STEP 4
                    360 LOCATE j,i:PRINT ag1$
                    370 LOCATE j,i+1:PRINT ag2$
                    380 LOCATE j,i+2:PRINT ag3$
                    390 NEXT:NEXT
```



```
400 FOR i=1 TO 3: LOCATE 1,i:PRINT SPACE$(36):NEXT
410 LOCATE 14,2:PEN 3:PRINT "Roger ROSEC"
420 SOUND 1,100,100,7,1,1
430 FOR i=5 TO 24:LOCATE 1,i:PRINT SPACE$(36):NEXT
440 LOCATE 18,5:PEN 1:PRINT "ET"
450 at$="AMSTRAD MAGAZINE"
460 ap1=277-LEN(at$)*16:INK 2,16
470 LOCATE 1,13:PEN 2:PRINT at$;:PEN 1
480 ap2=ap1:ap3=280:ap4=206
490 FOR i=1 TO 7:ap5=0:SOUND 1,100,150,10,2,0,2:FOR j=1 TO LEN(at$)*8
500 IF TEST(ap5,ap4)=2 THEN PLOT ap1,ap3,1:PLOT ap1,ap3-2:PLOT ap1+2,ap3:PLOT ap1+2,ap3-2
510 ap1=ap1+4:ap5=ap5+2:NEXT
520 ap3=ap3-4:ap4=ap4-2:ap1=ap2:NEXT
530 LOCATE 1,13:PRINT SPACE$(LEN(at$))
540 INK 2,3
550 LOCATE 1,14:PEN 2:PRINT "VOUS PRESENTENT"
560 AA$(1)="4115000000000004110000000000010000000010"
570 AA$(2)="1000041150141010001001011150104104115010"
580 AA$(3)="1000000010100010001001010010142001001010"
590 AA$(4)="1015041110100011001001010010110001112010"
600 AA$(5)="1001010010100010001001010010135001000010"
610 AA$(6)="3112031110100010003112010010103103112031"
620 WINDOW #1,1,40,17,24:PAPER #1,2:CLS#1
630 FOR AU=1 TO 6
640 FOR AI=1 TO 40
650  IF MID$(AA$(AU),AI,1)="0" THEN ta$(au)=ta$(au)+CHR$(32) :GOTO 700
660 IF MID$(AA$(AU),AI,1)="1" THEN ta$(au)=ta$(au)+CHR$(143):GOTO 700
670 IF MID$(AA$(AU),AI,1)="2" THEN ta$(au)=ta$(au)+CHR$(212):GOTO 700
680 IF MID$(AA$(AU),AI,1)="3" THEN ta$(au)=ta$(au)+CHR$(213):GOTO 700
690 IF MID$(AA$(AU),AI,1)="4" THEN ta$(au)=ta$(au)+CHR$(214) ELSE ta$(au)=ta$(au)+CHR$(215)
700 NEXT AI :NEXT au
710 LOCATE 1,24:PRINT STRING$(40,CHR$(128))
720 PEN 1:FOR au=6 TO 1 STEP-1:LOCATE 1,AU+17:PRINT ta$(AU):NEXT AU
730 LOCATE 1,17:PRINT STRING$(40,CHR$(128))
740 REM Si vous sauvez sur disquette, ne    tapez pas la ligne suivante
750 PEN 3:LOCATE 6,25:PRINT "Patience ! Chargement en cours."
760 REM Si vous sauvez sur disquette, ne    tapez pas la ligne suivante
780 REM -- FIN DU PREMIER PROGRAMME ---
790 REM --------------------------------
```

```
800

810

1000 REM ********************************

1010 REM ** DEBUT DEUXIEME PROGRAMME ***

1020 REM ********************************

1030 REM Enregistrez ce programme a la          suite de l'autre en lui donnant le nom
indique ligne 770

1040 ON BREAK GOSUB 3770

1050 LOCATE 1,25: PRINT SPACE$(25)

1060 FOR p=1 TO 17:PRINT

1070 NEXT p

1080 DIM r$(20):DIM ha(100):DIM jj$(3)

1090 REM ******************************

1100 REM ***** data musique ***********

1110 DATA 262,294,330,262,262,294,330,262,330,349,392,392,330,349,392,392,392,440,392,349,
330,330,262,262,392,440,392,349,330,330,26
2,262,262,262,196,196,262,262,262,262,262,196,196,262,262,262

1120 REM

1130 IF man<>0 THEN PEN 1:FOR i=1 TO 6:LOCATE 1,i:PRINT ta$(i):NEXT

1140 REM

1150 j1$=CHR$(240)

1160 j2$=CHR$(241)

1170 j3$=CHR$(242)

1180 j4$=CHR$(243)

1190 j5$=CHR$(224)

1200 man=2

1210 INK 2,11

1220 FOR i=1 TO 6:WINDOW #i,2,39,i*3+5,i*3+8:IF INT(i/2)=i/2 THEN PAPER #i,1:PEN #i,2 ELSE
PAPER #i,2:PEN #i,1

1230 CLS #i:NEXT

1240 LOCATE #1,1,2:PRINT #1," -  Regles du jeu . . . . . . . . . ";:PEN #1,3:PRINT #1,"1  "

1250 LOCATE #2,1,2:PRINT #2," -  Utilisation du curseur  . . . ";:PEN #2,3:PRINT #2,"2  "

1260 LOCATE #3,1,2:PRINT #3," -  Utilisation du joystick . . . ";:PEN #3,3:PRINT #3,"3  "

1270 LOCATE #4,1,2:PRINT #4," -  Redefinition des touches  . . ";:PEN #4,3:PRINT #4,"4  "

1280 LOCATE #5,1,2:PRINT #5," -  Jeu avec un seul joueur   . . ";:PEN #5,3:PRINT #5,"5  "

1290 LOCATE #6,1,2:PRINT #6," -  Jeu a deux joueurs  . . . . . ";:PEN #6,3:PRINT #6,"6  "

1300 LOCATE #man,37,2:PEN #man,3:PRINT #man,"*"

1310 RESTORE 1110

1320 READ m:SOUND 1,m/2,20,5 :SOUND 2,m*0.75,20,3:SOUND 4,m,20,5

1330 IF INKEY(64)<>-1 OR INKEY(13)<>-1 THEN 3800
```



```
1340 IF INKEY(65)<>-1 OR INKEY(14)<>-1 THEN 1140
1350 IF INKEY(57)<>-1 OR INKEY(5)<>-1 THEN 1400
1360 IF INKEY(56)<>-1 OR INKEY(20)<>-1 THEN 1480
1370 IF INKEY(49)<>-1 OR INKEY(12)<>-1 THEN 1890
1380 IF INKEY(48)<>-1 OR INKEY(4)<>-1 THEN 1700
1390 mu=mu+1:IF mu=45 THEN mu=0:GOTO 1310 ELSE 1320
1400 REM ***** JEU AVEC JOYSTICK ******
1410 j1$=CHR$(11)
1420 j2$=CHR$(10)
1430 j3$=CHR$(8)
1440 j4$=CHR$(9)
1450 j5$=CHR$(224)
1460 man=3
1470 GOTO 1210
1480 REM ** REDEFINITION DES TOUCHES **
1490 FOR i=8 TO 25:LOCATE 1,i:PRINT SPACE$(39)
1500 NEXT
1510 INK 2,3
1520 LOCATE 9,10: PEN 3:PRINT "REDEFINIION DES TOUCHES"  :PEN 2
1530 FOR i=1 TO 10:c$=INKEY$:NEXT
1540 LOCATE 15,14:PRINT "Haut   ?  ";
1550 GOSUB 1660:j1$=b$
1560 LOCATE 15,16:PRINT "Bas    ?  ";
1570 GOSUB 1660:j2$=b$
1580 LOCATE 15,18:PRINT "Gauche ?  ";
1590 GOSUB 1660:j3$=b$
1600 LOCATE 15,20:PRINT "Droite ?  ";
1610 GOSUB 1660:j4$=b$
1620 LOCATE 15,22:PRINT "Repetition ?  ";
1630 GOSUB 1660:j5$=b$
1640 FOR i=8 TO 24:LOCATE 1,i:PRINT SPACE$(39):NEXT
1650 man=4:GOTO 1210
1660 REM --- Valeur des touches -----
1670 b$=INKEY$:IF b$<>"" THEN PRINT b$ ELSE 1660
1680 SOUND 1,2056,10,7
1690 RETURN
1700 REM ****** JEU A 2 JOUEURS *******
1710 jb=2
1720 FOR i=8 TO 25:LOCATE 1,i:PRINT SPACE$(39):NEXT
```

```
1730 INK 2,3
1740 LOCATE 9,11: PEN 3:PRINT "JEU A DEUX JOUEURS":PEN 2
1750 FOR i=1 TO 10:c$=INKEY$:NEXT
1760 LOCATE 1,13:INPUT "Joueur 1 ";nm1$
1770 LOCATE 1,15:INPUT "Joueur 2 ";nm2$
1780 LOCATE 1,17:PRINT "Qui commence ? :"
1790 LOCATE 1,19:PRINT " 1 - ";nm1$
1800 LOCATE 1,20:PRINT " 2 - ";nm2$
1810 LOCATE 1,21:PRINT " 3 - Au hazard  ":nx$=INKEY$:IF nx$="" THEN 1810
1820 IF nx$="1" THEN jj$(1)=LEFT$(nm1$,6):jj$(2)=LEFT$(nm2$,6):GOTO 1860
1830 IF nx$="2" THEN jj$(1)=LEFT$(nm2$,6):jj$(2)=LEFT$(nm1$,6):GOTO 1860
1840 IF nx$="3" THEN jm=RND*10:IF jm<5 THEN nx$="1":GOTO 1820 ELSE nx$="2":GOTO 1830
1850 GOTO 1780
1860 jj$(3)=jj$(1):tr=1:LOCATE 1,23:PEN 3:PRINT "C'est ";jj$(1);" qui commence."
1870 FOR i=1 TO 3000:NEXT
1880 GOTO 1940
1890 REM ******* JEU 1 JOUEUR *********
1900 jb=1
1910 REM ******* DEBUT DU JEU *********
1920 REM
1930 REM ******* TRACE DU JEU *********
1940 MODE 0
1950 RANDOMIZE TIME
1960 INK 4,16:INK 5,11
1970 SYMBOL 200,206,238,168,168,172,236,200,168
1980 SYMBOL 201,68,238,170,138,138,138,138,170
1990 SYMBOL 202,204,238,170,170,170,234,202,170
2000 SYMBOL 203,174,174,0,0,255,255,0,0
2010 SYMBOL 204,238,68,0,0,255,255,0,0
2020 SYMBOL 205,174,172,0,0,254,254,0,0
2030 SYMBOL 206,2,7,5,4,6,3,1,5
2040 SYMBOL 207,34,119,85,69,69,69,69,85
2050 SYMBOL 208,103,119,84,84,86,118,100,84
2060 SYMBOL 209,7,2,0,0,7,7,0,0
2070 SYMBOL 210,119,34,0,0,255,255,0,0
2080 SYMBOL 211,87,87,0,0,255,255,0,0
2090 SYMBOL 221,255,255,128,128,128,128,159,159
2100 SYMBOL 222,255,255,1,1,1,1,189,189
2110 SYMBOL 223,159,159,153,152,152,152,152,155
```



```
2120 SYMBOL 225,189,189,177,49,49,49,49,253
2130 SYMBOL 226,155,155,155,153,153,153,159,159
2140 SYMBOL 227,253,253,253,177,177,177,177,177
2150 SYMBOL 228,159,159,128,128,128,128,255,255
2160 SYMBOL 229,177,177,1,1,1,1,255,255
2170 SYMBOL 244,0,1,1,3,3,7,7,15
2180 SYMBOL 245,0,128,128,192,192,224,224,240
2190 SYMBOL 246,15,7,7,3,3,1,1,0
2200 SYMBOL 247,240,224,224,192,192,128,128,0
2210 SYMBOL 248,15,31,31,63,63,127,127,255
2220 SYMBOL 249,240,248,248,252,252,254,254,255
2230 SYMBOL 250,255,127,127,63,63,31,31,15
2240 SYMBOL 251,255,254,254,252,252,248,248,240
2250 SYMBOL 252,255,126,126,60,60,24,24,0
2260 SYMBOL 253,0,24,24,60,60,126,126,255
2270 INK 10,26:INK 11,24:INK 12,17:INK 13,6:INK 2,3
2280 RESTORE 3790
2290 FOR dg=1 TO 7
2300 READ dar,db,dt
2310 LOCATE dt,dg:PEN 10:PRINT CHR$(dar+240);STRING$(db,CHR$(143));CHR$(dar+241)
2320 IF dg>3 THEN dar=dar-4
2330 LOCATE dt,25-dg:PEN 11:PRINT CHR$(dar+242);STRING$(db,CHR$(143));CHR$(dar+243)
2340 NEXT dg
2350 LOCATE 9,8:PEN 10:PRINT CHR$(252):LOCATE 12,8:PRINT CHR$(252)
2360 LOCATE 9,17:PEN 11:PRINT CHR$(253):LOCATE 12,17:PRINT CHR$(253)
2370 FOR dv=6 TO 15 STEP 9
2380 IF dv=6 THEN PEN 12 ELSE PEN 13
2390 LOCATE dv,5:PRINT CHR$(253)
2400 LOCATE dv,20:PRINT CHR$(252)
2410 LOCATE dv-1,6:PRINT CHR$(244);CHR$(143);CHR$(245)
2420 LOCATE dv-1,19:PRINT CHR$(246);CHR$(143);CHR$(247)
2430 LOCATE dv-1,7:PRINT CHR$(248);CHR$(143);CHR$(249)
2440 LOCATE dv-1,18:PRINT CHR$(250);CHR$(143);CHR$(251)
2450 NEXT dv
2460 FOR dg2=8 TO 17
2470 READ dar2
2480 PEN 12:LOCATE 5,dg2:PRINT STRING$(3,CHR$(143))
2490 LOCATE 8,dg2:PRINT CHR$(dar2)
2500 PEN 13:LOCATE 14,dg2:PRINT STRING$(3,CHR$(143))
```

```
2510 IF dar2=133 THEN dar2=139
2520 LOCATE 13,dg2:PRINT CHR$(dar2-1)
2530 NEXT dg2
2540 za=10:zb=11:GOSUB 3270
2550 LOCATE 15,25:PEN 3:PRINT "ENTER"
2560 IF jb=2 THEN LOCATE 1,23:PEN 3:PRINT jj$(tr):LOCATE 1,25:PEN 2:PRINT "A vous"
2570 pp1$=CHR$(200)+CHR$(201)+CHR$(202)
2580 pp2$=CHR$(203)+CHR$(204)+CHR$(205)
2590 pp3$=CHR$(206)+CHR$(207)+CHR$(208)
2600 pp4$=CHR$(209)+CHR$(210)+CHR$(211)
2610 LOCATE 1,1:PEN 2:PRINT pp1$:LOCATE 1,2:PRINT pp2$
2620 LOCATE 18,1:PEN 2:PRINT pp3$:LOCATE 18,2:PRINT pp4$
2630 LOCATE 1,4:PRINT USING"###";st
2640 FOR i=1 TO 10:c$=INKEY$:NEXT
2650 ttt=0
2660 RESTORE 1110
2670 ttt=ttt+1:IF ttt>500 THEN READ m:SOUND 1,m,20,5 :SOUND 2,m*2,20,3:SOUND 4,m/2,20,5
2680 IF INKEY(18)<>-1 THEN 2700
2690 mu=mu+1:IF mu=45 THEN mu=0:GOTO 2660 ELSE 2670
2700 hw=1:hj=1:jr=1
2710 LOCATE 15,25:PRINT "      "
2720 FOR i=1 TO 1500:NEXT
2730 tt=3:FOR i=1 TO 100:ha(i)=0:NEXT
2740 PEN 0:GOSUB 3320
2750 REM
2760 tp=50-INT(10*LOG(hw))
2770 IF jr=1 THEN INK 0,11:BORDER 11 ELSE INK 0,16:BORDER 16
2780 LOCATE 1,23:PRINT "       ":PEN 3:LOCATE 1,23:IF jb=2 AND jr=1 THEN PRINT jj$(tr) ELSE
IF jb =2 THEN PRINT jj$(tr+1)
2790 IF jr=1 AND jb=1 THEN ha(hw)=INT(RND*4)+1:LOCATE 1,25:PRINT "Ecoutez":GOSUB 3190
2800 IF jb=1 THEN jr=2
2810 PEN 1:LOCATE 1,25:PRINT "A vous "
2820 IF jr=1 THEN INK 0,11:BORDER 11 ELSE INK 0,16:BORDER 16
2830 tt=hw+2:LOCATE 9,12:PEN 2:PRINT USING "###";tt
2840 FOR hj=1 TO hw
2850 EVERY 50,1 GOSUB 3120
2860 FOR u=1 TO 10:c$=INKEY$:NEXT
2870 hh$=INKEY$: IF tt<1 THEN 3360 ELSE IF hh$="" THEN 2870
2880 note=0
2890 IF hh$=j1$ THEN note=1:GOTO 2950
```



```
2900 IF hh$=j2$ THEN note=2:GOTO 2950
2910 IF hh$=j3$ THEN note=3:GOTO 2950
2920 IF hh$=j4$ THEN note=4:GOTO 2950
2930 IF hh$=j5$ THEN GOSUB 3190
2940 GOTO 2870
2950 IF ha(hj)=0 THEN ha(hj)=note
2960 IF note<>ha(hj) THEN 3360
2970 ON ha(hj) GOSUB 3040,3060,3080,3100
2980 NEXT hj
2990 PEN 2
3000 IF jr=1 THEN jr=2 ELSE jr=1
3010 sc=hw*10:LOCATE 18,4:PRINT USING"###";sc
3020 FOR del=1 TO 1000:NEXT
3030 hw=hw+1:GOTO 2750
3040 REM ***** LES NOTES **************
3050 za=10:zb=2:GOSUB 3270:SOUND 1,358,tp,7:FOR i=1 TO 300:NEXT:PEN 10:GOSUB 3320:RETURN
3060 REM
3070 za=10:zb=20:GOSUB 3270:SOUND 1,284,tp,7:FOR i=1 TO 300:NEXT:PEN 11:GOSUB 3320:RETURN
3080 REM
3090 za=6:zb=11:GOSUB 3270:SOUND 1,213,tp,7:FOR i=1 TO 300:NEXT:PEN 12:GOSUB 3320:RETURN
3100 REM
3110 za=14:zb=11:GOSUB 3270:SOUND 1,190,tp,7:FOR i=1 TO 300:NEXT:PEN 13:GOSUB 3320:RETURN
3120 REM ***** HORLOGE ****************
3130 IF jr=1 AND jb=1 THEN LOCATE 9,12:PRINT "   ":GOTO 3180
3140 tt=tt-1
3150 SOUND 1,284,5,7
3160 PEN 2
3170 LOCATE 9,12:PRINT USING "###";tt
3180 RETURN
3190 REM ****** REPETITION ************
3200 FOR ji=1 TO hw
3210 ON ha(ji) GOSUB 3040,3060,3080,3100
3220 FOR yy=1 TO 400:NEXT
3230 NEXT ji
3240 PEN 2
3250 RETURN
3260 REM ******* NOTE JOUEE ***********
3270 DI:PEN 1:LOCATE za,zb:PRINT CHR$(221);CHR$(222)
3280 LOCATE za,zb+1:PRINT CHR$(223);CHR$(225)
```

```
3290 LOCATE za,zb+2:PRINT CHR$(226);CHR$(227)
3300 LOCATE za,zb+3:PRINT CHR$(228);CHR$(229)
3310 EI:RETURN
3320 REM ******* EFFACEMENT ***********
3330 DI:FOR zc=0 TO 3:LOCATE za,zb+zc:PRINT CHR$(143);CHR$(143):NEXT zc :EI
3340 PEN 2
3350 RETURN
3360 REM ********* PERDU **************
3370 tt=0
3380 SOUND 1,100,200,6,2,1
3390 LOCATE 9,12:PRINT REMAIN(1)
3400 IF sc>st THEN st=sc
3410 FOR gf=1 TO 25:LOCATE 1,gf:PRINT "    ":NEXT
3420 LOCATE 9,12:PRINT "   "
3430 LOCATE 1,25:PRINT "       "
3440 LOCATE 1,23:PRINT "      "
3450 OUT &BC00,13
3460 FOR ghj=1 TO 8
3470 OUT &BD00,ghj
3480 FOR tepo=1 TO 40:NEXT tepo:NEXT ghj
3490 FOR i=1 TO 1500:NEXT:GOSUB 3200
3500 IF jb=1 GOTO 3610
3510 IF jr=2 THEN sc(tr)=sc ELSE sc(tr+1)=sc
3520 IF sc(1)>sc1 THEN sc1=sc(1)
3530 IF sc(3)>sc1 THEN sc1=sc(3)
3540 IF sc(2)>sc2 THEN sc2=sc(2)
3550 IF sc2>sc1 THEN scl2=9 :scl1=11 ELSE scl1=9:scl2=11
3560 PEN 1
3570 LOCATE 18,scl1:PRINT LEFT$(jj$(1),3)
3580 LOCATE 2,scl1+1:PRINT USING"###";sc1
3590 LOCATE 18,scl2:PRINT LEFT$(jj$(2),3)
3600 LOCATE 2,scl2+1:PRINT USING"###";sc2
3610 PEN 1
3620 LOCATE 2,2:PRINT pp1$
3630 LOCATE 2,3:PRINT pp2$
3640 PEN 3:LOCATE 2,5:PRINT USING"###";st
3650 PEN 2:LOCATE 1,16:PRINT "UNE"
3660 LOCATE 19,17:PRINT "AU"
3670 LOCATE 1,18:PRINT "TRE"
```



```
3680 LOCATE 1,20:PEN 3:PRINT "O/N"
3690 za=3:zb=22:GOSUB 3270
3700 ttt=0
3710 IF tr=1 THEN tr=2 ELSE tr=1
3720 RESTORE 1110
3730 IF INKEY(34)<>-1 THEN 1940
3740 IF INKEY(46)<>-1 THEN MODE 1:sc1=0:sc2=0:sc(1)=0:sc(2)=0:sc(3)=0:GOTO 1120
3750 ttt=ttt+1:IF ttt>500 THEN READ m:SOUND 1,m,20,5 :SOUND 2,m*2,20,3:SOUND 4,m/2,20,5
3760 mu=mu+1:IF mu=45 THEN mu=0:GOTO 3720 ELSE 3730
3770 MODE 1:PEN 1:END
3780 REM
3790 DATA 8,4,8,4,6,7,8,6,7,10,6,7,6,6,7,10,4,8,6,4,8,245,249,251,133,133,133,133,249,
251,247
3800 REM ***** regles du jeu *********
3810 r$(1)="              REGLES DU JEU"
3820 r$(2)=" "
3830 r$(3)="  Le but du jeu consiste a reproduire"
3840 r$(4)="une serie sonore de plus en plus longue"
3850 r$(5)="composee de quatre notes differentes,"
3860 r$(6)="qui sont associees a une couleur."
3870 r$(7)=" "
3880 r$(8)="  VERSION 1 JOUEUR  : La serie de notes"
3890 r$(9)="est generee par l'ordinateur."
3900 r$(10)=" "
3910 r$(11)="  VERSION 2 JOUEURS : Les adversaires "
3920 r$(12)="augmentent la serie d'une note a chaque"
3930 r$(13)="fois."
3940 r$(14)= " "
3950 r$(15)="En cours de jeu, pour faire repeter la"
3960 r$(16)="derniere sequence, tapez COPY "
3970 FOR ri=7 TO 25:LOCATE 1,ri:PRINT SPACE$(39):NEXT
3980 FOR ry=1 TO 16
3990 INK 2,3
4000 IF ry=1 THEN PEN 3 ELSE PEN 2
4010 IF ry=2 OR ry=7 OR ry=10 OR ry=14  THEN 4030
4020 LOCATE 1,ry+8:PRINT r$(ry):SOUND 1,100,20,15,0,1:FOR i=1 TO 500:NEXT
4030 NEXT ry
4040 FOR i=1 TO 10000:IF INKEY(18)=0 THEN i=10000 ELSE NEXT
4050 FOR ri=7 TO 25:LOCATE 1,ri:PRINT SPACE$(39):NEXT
4060 GOTO 1210
```

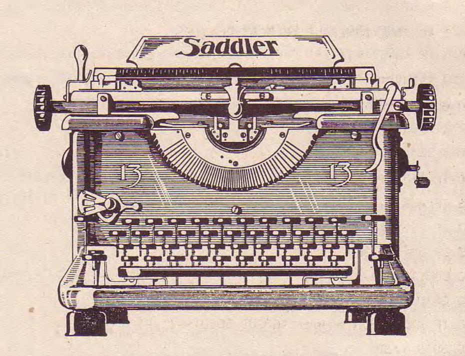
Traitement de texte

Une fois la fastidieuse opération d'entrée du programme effectuée, sauvez votre programme soit sur cassette, soit sur disquette. La compatibilité est alors automatique.

Faites "RUN" et vous verrez s'afficher à l'écran la question : "Désirez-vous reconfigurer l'imprimante ?" En effet il peut être nécessaire d'affecter les caractères accentués à d'autres codes ASCII que ceux prédéfinis dans le programme.

Si vous répondez oui, vous verrez alors s'afficher chaque caractère à l'écran et il vous faudra affecter à chacun de ces caractères la valeur adéquate. Pour cela reportez-vous au manuel de votre imprimante).

Vous accédez maintenant au menu. Les options suivantes vous sont alors proposées :



1) Chargement
Si vous sélectionnez cette option, le programme chargera d'une cassette, ou d'une disquette le fichier de texte préalablement sauvé. Le fichier est nommé dans tous les cas "texte".

2) Sauvegarde
Cette fonction est identique à celle de chargement en mode sauvegarde.

3) Effacement mémoire
Efface tout le fichier de texte contenu en mémoire. (Attention le format du texte sera également effacé).

4) Azerty
Passage en mode Azerty

5) Qwerty
Retour en mode Qwerty

6) Format des pages
Définit le nombre de pages et le nombre de colonnes du texte. (Cette option doit absolument être sélectionnée avant de taper du texte).

7) Copie sur imprimante
Copie le texte sur imprimante dans la largeur définie par l'option de format.

8) Accès au texte
Permet de passer au mode d'édition du texte.

Les caractères de contrôle :

DEL : efface tout caractère présent devant le curseur.

Les curseurs :
Ils permettent de se déplacer dans le texte.

ENTER : permet de passer à la ligne suivante directement, la ligne où est situé le curseur.

CTRL + c : retour au menu

Eric Charton

```
20 '
30 '      traitement de texte
40 '
45 '
50 OPENOUT "ID":CLOSEOUT
80 a=64:a1=92:a2=93:a3=94:a4=123:a5=124:a6=125
100 WIDTH 80
110 GOSUB 60000
115 MODE 2
120 INPUT "desirez vous reconfigurer l'imprimante  ",a$
125 IF a$="o" THEN GOSUB 62000
200 GOTO 30000
505 '
507          'INITIALISATION
510 '
520 X=1
```



```
530 Y=1
535 ec=1
540 CLS
550 MODE 2
1000 '
1010 ' ROUTINE GENERALE DE TEST CLAVIER
1020 '
1050 GOSUB 5000
1060 GOSUB 7000
1100 A$=INKEY$
1105 IF A$<>"" THEN IF ASC (A$)>239 THEN GOSUB 2000:GOTO 1100
1110 IF A$="" THEN 1100
1116 IF ASC (a$)=127 THEN GOSUB 2140:GOTO 1100
1117 IF  (a$)=CHR$ (13) THEN GOSUB 3000:GOTO 1100
1118 IF a$="" THEN 30000
1120 LOCATE Y,X:PRINT A$:tr$ (x+(24*ec),y)=a$: Y=Y+1:IF Y>LONG THEN Y=1:X=X+1
1130 GOSUB 7000
1150 IF x>24 THEN ec=ec+1:GOSUB 5000:x=1:y=1
1200 GOTO 1100
2000 ' saisie des caracteres de controle
2100 IF INKEY$=CHR$ (243) AND Y<long THEN Y=Y+1
2110 IF INKEY$=CHR$ (242) AND Y>1 THEN Y=Y-1
2120 IF INKEY$=CHR$ (240)  THEN X=X-1:IF x=0 THEN IF ec>1 THEN ec=ec-1:GOSUB 500
0:x=24:y=1 ELSE x=1
2130 IF INKEY$=CHR$ (241)  THEN X=X+1 :IF x=25 THEN CLS :ec=ec+1:GOSUB 5000:x=1:
y=1
2140 IF INKEY (79)=0 THEN tr$(x+(24*ec),y)="":IF Y>1 THEN y=y-1 ELSE IF X>1 THEN
 x=x-1:y=long:LOCATE 1,x+1:PRINT " "
2150 GOSUB 7000
2200 RETURN
3000 IF x<24 THEN LOCATE y,x:PRINT tr$(x+(24*ec),y):IF tr$(x+(24*ec),y)="" THEN
LOCATE y,x:PRINT " "
3005 IF x<24 THEN x=x+1:y=1
3010 GOSUB 7000
3100 RETURN
5000 '
5003 '     increment d'une page
5004 '
5010 IF ec=page+1 THEN y=long:x=24:ec=ec-1: RETURN
```

```
5015 CLS
5020 GOSUB 6000
5030 FOR t=1 TO 24
5040 FOR u=1 TO long
5050 LOCATE u,t
5060 PRINT tr$(t+(24*ec),u)
5070 NEXT u
5080 NEXT t
5090 RETURN
6000 '
6010 ' indications
6020 '
6030  LOCATE 1,25:PRINT "page no :";ec;"   largeur :";long;"   nombre de pages:"
;page
6100 RETURN
7000 '
7010 ' affichage curseur
7020 '
7040 IF y>1 THEN LOCATE y-1,x:IF tr$(x+(24*ec),y-1)="" THEN PRINT " " ELSE PRINT
 tr$(x+(24*ec),y-1)
7050 IF y<long THEN LOCATE y+1,x:IF tr$(x+(24*ec),y+1)="" THEN PRINT " " ELSE PR
INT tr$(x+(24*ec),y+1)
7055 IF x=0 OR y=0 THEN GOTO 55000
7060 IF x>1 THEN LOCATE y,x-1:IF tr$(x-1+(24*ec),y)="" THEN PRINT " " ELSE PRINT
 tr$(x-1+(24*ec),y)
7070 IF x<24 THEN LOCATE y,x+1:IF tr$(x+1+(24*ec),y)="" THEN PRINT " " ELSE PRIN
T tr$(x+1+(24*ec),y)
7100 PRINT CHR$ (22)+CHR$ (1)
7200 LOCATE y,x
7320 PRINT CHR$ (210)
7330 PRINT CHR$ (22)+CHR$ (0)
7400 RETURN
9075 IF TR$(I,J)="" THEN GOTO 59085
20000 '
30000 '
30010 '
30015 '      options
30030 '
30040 '
```



```
30045 ON ERROR GOTO 55000
30050 MODE 1
30060 LOCATE 10,2:PRINT "options
30070 LOCATE 3,5:PRINT "1 chargement":LOCATE 3,6:PRINT "2 sauvegarde":LOCATE 3,7

30080 PRINT "3 effacement memoire":LOCATE 3,8:PRINT"4 azerty":LOCATE 3,9:PRINT"5
 qwerty": LOCATE 3,10:PRINT "6 format des pages":LOCATE 3,11:PRINT "7 copie sur
imprimante":LOCATE 3,12:PRINT "8 acces au texte"
30199 a$=INKEY$
30200 IF a$="" THEN 30199
30210 IF a$="8"THEN 505
30215 IF a$="7"THEN 59000
30220 IF a$="6"THEN GOTO 32000
30225 IF a$="3"THEN CLEAR:GOTO 30000
30230 IF a$="4"THEN GOSUB 50000 :PRINT "azerty OK":FOR i=1 TO 1000:NEXT:GOTO 300
00
30240 IF a$="5"THEN GOSUB 52000 :PRINT "qwerty OK":FOR i=1 TO 1000:NEXT:GOTO 300
00
30250 IF a$="1" THEN GOTO 56000
30260 IF a$="2" THEN GOTO 57000
31999 GOTO 30199
32000 LOCATE 2,23
32005 CLEAR
32010 INPUT "nombre de pages (1 page=24 lignes)?",page
32100 INPUT "largeur d'une page (10/80)",long
32105 DIM tr$((page+1)*25,long+1)
32110 GOTO 30000
50000 '
50010 '
50020 '     azerty
50030 '
50040 '
51000 KEY DEF 67,0,97,65:REM transforme q en a
51100 KEY DEF 59,0,122,90:REM transforme w en z
51200 KEY DEF 69,0,113,81:REM transforme a en q
51300 KEY DEF 71,0,119,87:REM transforme z en w
51400 KEY DEF 29,0,109,77:REM transforme : en m
51500 KEY DEF 38,0,58,42:REM transforme m en :
51600 RETURN
```



```
52000 '
52100 '
52200 '     qwerty
52210 '
52220 '
52300 KEY DEF 67,0,113,81:REM transforme a en q
52310 KEY DEF 59,0,119,87:REM transforme z en w
52320 KEY DEF 69,0,97,65:REM transforme q en a
52330 KEY DEF 71,0,122,90:REM transforme w en z
52340 KEY DEF 29,0,58,42:REM transforme m en :
52350 KEY DEF 38,0,109,77:REM transforme : en m
53000 RETURN
55000 '
55010 '
55020 '  tableau non defini
55030 '
55040 '
55050 MODE 1
55060 LOCATE 24,5
55070 PRINT "format des feuillets  non defini erreur"
55075 IF INKEY$="" THEN 55075
55080 GOTO 30000
56000 '
56010 ' CHARGEMENT
56020 '
56025 CLEAR
56030 OPENIN "texte
56040 INPUT #9,page:INPUT #9,long
56045 DIM tr$((page+1)*24,long )
56050 FOR I =0 TO ((PAGE+1)*24):FOR J=0 TO LONG
56060 LINE INPUT #9,tr$ (i,j)
56070 NEXT j:NEXT i
56080 CLOSEIN
56090 GOTO 30000
57000 '
57010 ' SAUVEGARDE
57020 '
57030 OPENOUT "TEXTE
57035 PRINT #9,page:PRINT #9,long
57040 FOR I =0 TO(( PAGE+1)*24):FOR J=0 TO LONG
57050 PRINT #9,tr$ (i,j)
57060 NEXT j:NEXT i
57065 CLOSEOUT
57070 GOTO 30000
59000 '
59010 '
59020 '
59030 '  sortie sur listing
59040 '
59050 '
59060 '
59070 FOR I =25 TO(24*(PAGE+1)):FOR J=1 TO LONG
59075 IF TR$(I,J)="" THEN PRINT #8," ";:GOTO 59085
59080 PRINT #8,tr$(I,J);
59085  NEXT j:PRINT #8,CHR$(13):NEXT i
59090 GOTO 30000
60000 '
60010 '
60020 '    caracteres accentues
60030 '
60040 '
60100 SYMBOL AFTER 64
60110 SYMBOL a,&60,&10,&78,&C,&7C,&CC,&76
60120 SYMBOL a1,&0,&0,&3C,&66,&60,&3E,&8,&18
60130 SYMBOL a2,&1C,&32,&38,&6C,&38,&10,&90,&60
60140 SYMBOL a3,&18,&24
60150 SYMBOL a4,&6,&8,&3C,&66,&7E,&60,&3C
60160 SYMBOL a5,&30,&8,&66,&66,&66,&66,&3E
60170 SYMBOL a6,&60,&10,&3C,&66,&7E,&60,&3C
60180 KEY 128,CHR$(a6)
60190 KEY 129,CHR$(a)
60200 KEY 131,CHR$(a2)
60210 KEY 132,CHR$(a1)
60220 KEY 134,CHR$(a3)
60230 KEY 135,CHR$(a4)
60240 KEY 137,CHR$(a5)
60250 KEY 136,CHR$(32)
60255 KEY 133,CHR$(32)
60260 KEY 130,CHR$(32)
60300 RETURN
62000 CLS
62005 PRINT CHR$(a);:INPUT ": correspondance ";a
62010 PRINT CHR$(a1);:INPUT ": correspondance ";a1
62020 PRINT CHR$(a2);:INPUT ": correspondance ";a2
62030 PRINT CHR$(a3);:INPUT ": correspondance ";a3
62040 PRINT CHR$(a4);:INPUT ": correspondance ";a4
62050 PRINT CHR$(a5);:INPUT ": correspondance ";a5
62060 PRINT CHR$(a6);:INPUT ": correspondance ";a6
62100 GOSUB 60000
63000 GOTO 30000
```